CONSTITUTIVE MODELING OF METAL MATRIX COMPOSITES UNDER CYCLIC/FATIGUE LOADS
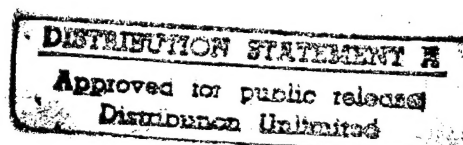
Final Report AFOSR-TR-96

Submitted to the
Air Force Office of Scientific Research

By

Professor and Principal Investigator:
Dr. George Z. Voyiadjis

Research Associates:
Dr. Taehyo Park
Dr. Ganesh Thiagarajan

Department of Civil and Environmental Engineering
Louisiana State University
Baton Rouge, LA 70803

LSU

March 1996

i

DISCLAIMER NOTICE

THIS DOCUMENT IS BEST
QUALITY AVAILABLE. THE
COPY FURNISHED TO DTIC
CONTAINED A SIGNIFICANT
NUMBER OF PAGES WHICH DO
NOT REPRODUCE LEGIBLY.

CONSTITUTIVE MODELING OF METAL MATRIX COMPOSITES UNDER CYCLIC/FATIGUE LOADS

Final Report

AFOSR Grant Nos. F49620-93-1-0097def and F49620-92-J-0463

By

Professor and Principal Investigator:
Dr. George Z. Voyiadjis

Research Associates:
Dr. Taehyo Park
Dr. Ganesh Thiagarajan

Department of Civil and Environmental Engineering
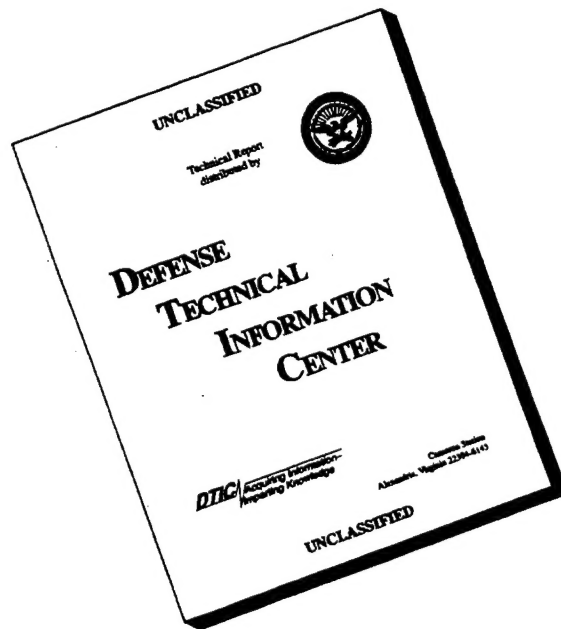Louisiana State University
Baton Rouge, LA 70803

March 1996

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information n estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	March 11, 1996	Final (1-1-93) – (12-31-95)

4. TITLE AND SUBTITLE	5. FUNDING NUMBERS
Constitutive Modeling of Metal Matrix Composites Using Cyclic Loads	F49620-93-1-0097DEF

6. AUTHOR(S)

George Z. Voyiadjis, Taehyo Park and
Ganesh Thiagarajan

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	8. PERFORMING ORGANIZATION REPORT NUMBER
Department of Civil and Environmental Engineering Louisiana State University Baton Rouge, LA 70803	None

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSORING/MONITORING AGENCY REPORT NUMBER
AFOSR/NA Building 410 Bolling AFB,DC 20332-6448 NA	93-1-0097

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT	12b. DISTRIBUTION CODE
Approved for public release; Distribution Unlimited	

13. ABSTRACT (Maximum 200 words)

A micromechanical consistent and systematic theory is developed for the analysis of damage mechanisms in metal matrix composites. A coupled incremental damage and plasticity theory of metal matrix composites is introduced here. This allows damage to be path dependent either on the stress history or the thermodynamic force conjugate to damage. This is achieved through the use of incremental damage tensors. The model involves both overall and local approaches to characterize damage in these materials due to cyclic and monotonic loads. Expressions are derived here for the elasto-plastic stress and strain concentration tensors for fibrous metal matrix composite materials in the damaged configurations.

A finite element analysis is used for quantifying each type of damage and predicting the failure loads of dog-bone shaped specimen and center-cracked laminate metal matrix composite plates. The development of damage zones and the stress-strain response are shown for two types of laminated layups.

14. SUBJECT TERMS		15. NUMBER OF PAGES
Metal matrix composites, cyclic damage, cyclic plasticity, damage, debonding, Finite Element		142
		16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	U

ACKNOWLEDGMENTS

Contents

List of Tables

List of Figures

PUBLICATIONS

The following is a list of the publications and other activities that resulted from the research work conducted for these two projects. They include four edited books, seventeen refereed journal articles, six book articles, and several conference proceedings and presentations. Numerous sessions were organized by Dr. Voyiadjis in conferences as a result of this project. This work also resulted in the supervision and completion of three Ph.D candidates and one M.S. candidate.

BOOKS

1. Voyiadjis, G. Z., and Allen, D. H., editors, Studies in Applied Mechanics, Damage and Interfacial Debonding in Composites, Elsevier, Amsterdam (to appear in 1996).

2. Voyiadjis, G. Z., and Ju, J. W., editors, Studies in Applied Mechanics, Vol. 41, Inelasticity and Micromechanics of Metal Matrix Composites, 351 p., Elsevier, Amsterdam, 1994.

3. Voyiadjis, G. Z., Bank, L. C., and Jacobs, L. J., editors, Studies in Applied Mechanics, Vol. 35, Mechanics of Materials and Structures, 436 p., Elsevier, Amsterdam, 1994.

4. Voyiadjis, G. Z., editor, Studies in Applied Mechanics, Vol. 34, Damage in Composite Materials, 286 p., Elsevier, Amsterdam, 1993.

SCIENTIFIC PAPERS
(a) Refereed Journal Articles

1. Voyiadjis, G. Z., and Park, T., "Local and Interfacial Damage Analysis of Metal Matrix Composites Using the Finite Element Method," International Journal of Engineering Fracture Mechanics, 52 manuscript pages (accepted for publication).

2. Voyiadjis, G. Z., and Zakaria Guelzim, "A Coupled Incremental Damage and Plasticity Theory for Metal Matrix Composites," Journal of the Mechanical Behavior of Materials, 28 manuscript pages (in press).

3. Voyiadjis, G. Z., and Thiagarajan, G., "A Cyclic Anisotropic Plasticity Model for Metal Matrix Composites," International Journal of Plasticity, 45 manuscript pages (accepted for publication).

4. Park, T., and Voyiadjis, G. Z., "Damage Analysis and Elasto-Plastic Behavior of Metal Matrix Composites Using the Finite Element Method," International Journal of Engineering Fracture Mechanics, 42 manuscript pages (accepted for publication).

5. Voyiadjis, G. Z., and Park, T., "Anisotropic Damage for the Characterization of the Onset of Macro-Crack Initiation in Metals," International Journal of Damage Mechancis, 38 manuscript pages (in press).

6. Kattan, P. I., and Voyiadjis, G. Z., "Damage-Plasticity in a Uniaxially-Loaded Composite Lamina: Overall Analysis," International Journal of Solids and Structures, Vol. 33, No. 4, February 1996, pp. 555-576.

7. Voyiadjis, G. Z., and Thiagarajan, G., "An Anisotropic Yield Surface Model for Directionally Reinforced Metal Matrix Composites," International Journal of Plasticity, Vol. 11, No. 8, 1995, pp. 867-894.

8. Voyiadjis, G. Z., Venson, A. R., and Kattan, P. I., "Experimental Determination of Damage Parameters in Uniaxially-Loaded Metal Matrix Composites Using the Overall Approach," International Journal of Plasticity, Vol. 11, No. 8, 1995 , pp. 895-926.

9. Voyiadjis, G. Z., and Park, T., "Anisotropic Damage of Fiber Reinforced MMC Using an Overall Damage Analysis," Journal of Engineering Mechanics, ASCE, Vol. 121, No. 11, 1995, pp. 1209-1217.

10. Voyiadjis, G. Z., and Venson, A. R., "Experimental Damage Investigation of a SiC-Ti Aluminide Metal Matrix Composite," International Journal of Damage Mechanics, Vol. 4, No. 4, October 1995, pp. 338-361.

11. Voyiadjis, G. Z., and Park T., "Local and Interfacial Damage Analysis of Metal Matrix Composites, " International Journal of Engineering Science, Vol. 33, No. 11, 1995, pp. 1595-1621.

12. Kattan, P. I., and Voyiadjis, G. Z., "Overall Damage and Elasto-Plastic Deformation in Fibrous Metal Matrix Composites," International Journal of Plasticity, Vol. 9, No. 8, 1993, pp. 931-949.

13. Voyiadjis, G. Z., and Kattan, P. I., "Damage of Fiber-Reinforced Composite Materials with Micromechanical Characterization," International Journal of Solids and Structures, Vol. 30, No. 20, 1993, pp. 2757-2778.

14. Shi, G., and Voyiadjis, G. Z., "A Computational Model for FE Ductile Plastic Damage Analysis of Plate Bending," Journal of Applied Mechanics, ASME, Vol. 60, No. 3, 1993, pp. 749-758.

15. Voyiadjis, G. Z., and Kattan, P. I., "Local Approach to Damage in Elasto-Plastic Metal Matrix Composites," International Journal of Damage Mechanics, Vol. 2, No. 1, 1993, pp. 92-114.

16. Kattan, P. I., and Voyiadjis, G. Z., "A Plasticity-Damage Theory for Large Deformation of Solids, Part II: Applications to Finite Simple Shear," International Journal of Engineering Science, Vol. 31, No. 1, 1993, pp. 183-199.

17. Kattan, P. I., and Voyiadjis, G. Z., "Micromechanical Modeling of Damage in Uniaxially Loaded Unidirectional Fiber-Reinforced Composite Laminae," International Journal of Solids and Structures, Vol. 30, No. 1, 1993, pp. 19-36.

(b) Refereed Proceedings

1. Voyiadjis, G. Z., and Guelzim, Z., "Incremental Damage Theory for Metal Matrix Composites," Contemporary Research in Engineering Science, edited by R. C. Batra, 1995, pp. 576-592.

2. Voyiadjis, G. Z., and Park, T., "Finite Element Analysis of Damage and Elasto-Plastic Behavior of Metal Matrix Composites," Computational Mechanics, Vol. 1, edited by S. N. Atluri, G. Yagawa and T. A. Cruse, Springer Publishing Company, 1995, p p. 1205-1210.

3. Voyiadjis, G. Z., and Thiagarajan, G., "A Damage Cyclic Plasticity Model for Metal Matrix Composites," Constitutive laws: Experiments and Numerical Implementation, edited by A. M. Rajendran and R. C. Batra, a publication of Interna tional Center for Numerical Methods in Engineering (CIMNE), 1995, pp. 159-170.

4. Voyiadjis, G. Z., and Thiagarajan, G., "A Cyclic Plasticity Model for Metal Matrix Composites," Computational Mechanics, Vol. 2, edited by S. N. Atluri, G. Yagawa and T. A. Cruse, Springer Publishing Company, 1995, pp. 1739-1744.

5. Voyiadjis, G. Z., and Venson, A. R., "Finite Element Implementation of the Overall Approach to Damage in Metal Matrix Composites," Engineering Mechanics, Proceedings of 10th Conference, Vol. 1, edited by Stein Sture, Engineering Mechanics Division of the ASCE, 1995, pp. 389-392.

6. Voyiadjis, G. Z., and Park, T., "Interfacial and Local Damage Analysis in Metal Matrix Composites," Damage Mechancis in Composites, 1994, edited by D.H. Allen, and J.W. Ju, ASME, AMD-Vol. 185, 1994, pp. 87-108.

7. Voyiadjis, G. Z., and Kattan, P. I., "Coupling of Damage and Viscoplasticity for Large Deformation of Metals," Large Plastic Deformations: Fundamental Aspects and Applications to Metal Forming, edited by C. Teodosiu, J. L. Raphanel, and F. Sidoroff, A. A. Balkema Publishing Company, 1993, pp. 345-352.

8. Voyiadjis, G. Z., Kattan, P. I., and Venson, A. R., "Evaluation of a Damage Tensor for Metal Matrix Composites," Direction des Etudes et Recherches d' Electricite de France: MECAMAT 93, International Seminar on Micromechanics of Materials, Vol. 84, 1993, pp. 406-417.

9. Voyiadjis, G. Z., and Kattan, P. I., "Micromechanical Characterization of Damage-Plasticity in Metal Matrix Composites," Studies in Applied Mechanics, Vol. 34, Damage in Composite Materials, edited by G. Z. Voyiadjis, Elsevier Publishing Company, 1993, pp. 67-102.

(e) Technical Reports

1. Voyiadjis, G. Z., Kattan, P. I., Venson, A. R., Park, T., and Thiagarajan, G., "Constitutive Modeling of Metal Matrix Composites Under Cyclic Loads - Second Annual Report (1993-94), F49620-93-1-0097DEF and F49620-92-J-0463," Air Force Office of Scientific Research, Washington, D.C., September 1994, 155 pages.

2. Voyiadjis, G. Z., Kattan, P. I., and Venson, A. R., "Constitutive Modeling and Finite Element Analysis of Fatigue Damage in Metal Matrix Composites - First Annual Report (1992-93), F49620-93-1-0097DEF and F49620-92-J-0463," Air Forc e Office of Scientific Research, Washington, D.C., October 1993, 130 pages.

PAPERS PRESENTED IN CONFERENCES AND INVITED LECTURES

1. Voyiadjis, G. Z., and Deliktas, B., "Damage in Metal Matrix Composites Using the Generalized Cells Method." To be presented at the 1996 ASME Mechanics and Materials Conference, Baltimore, Maryland, June 1996.

2. Voyiadjis, G. Z., and Echle, R., "Fatigue Model for Damage in Metal Matrix Composites." Invited lecture to be presented in the Symposium on "Application of Continuum Damage Mechanics to Fatigue and Fracture," at the ASTM Conference, Orlando, Florida, May 1996.

3. Voyiadjis, G. Z., and Park, T., "Effect of Damage Due to Delamination on the Flexural Stiffness of Laminated Composite Plates." Invited lecture presented at the Symposium on "Damage and Interfacial Debonding in Composites," at the 32nd Annual Technical Meeting of the Society of Engineering Science, New Orleans, Louisiana, October 1995.

4. Thiagarajan, G., and Voyiadjis, G. Z., "Cyclic Damage with Cyclic Plasticity in Metal Matrix Composites." Lecture presented at the 32nd Annual Technical Meeting of the Society of Engineering Science, New Orleans, Louisiana, October 1995.

5. Voyiadjis, G. Z., and Guelzim, Z., "Incremental Damage Theory for Metal Matrix Composites." Invited lecture presented at the "Eringen Medal Symposium: S. Atluri," at the 32nd Annual Technical Meeting of the Society of Engineering Science, New Orleans, Louisiana, October 1995.

6. Park, T., and Voyiadjis, G. Z., "Analysis of Interfacial and Local Damage in Metal Matrix Composites using the FE Method." Lecture presented at the 32nd Annual Technical Meeting of the Society of Engineering Science, New Orleans, Louisiana, October 1995.

7. Thiagarajan, G., and Voyiadjis, G. Z., "Cyclic Plasticity with Damage in Metal Matrix Composites." Lecture presented at the Symposium on "Micromechanics of Fibers or Composites," at the Second International Conference on Composites Engineering, New Orleans, Louisiana, August 1995.

8. Voyiadjis, G. Z., and Rainer, E., "Fatigue Damage in Metal Matrix Composites." Invited lecture presented at the Symposium on "Fatigue of Composites," at the Second International Conference on Composites Engineering, New Orleans, Louisiana, August 1995.

9. Park, T., and Voyiadjis, G. Z., "Stress and Strain Concentration Tensors for Interfacial Damage." Lecture presented at the Symposium on "Micromechanics of Fibers or Composites," at the Second International Conference on Composites Engineering, New Orleans, Louisiana, August 1995.

10. Venson, A. R., and Voyiadjis, G. Z., "Experimental Determination of Damage in Titanium Aluminide with Silicon Carbide Fibers." Lecture presented at the Symposium on "Metal Matrix Composites," at the Second International Conference on Composites Engineering, New Orleans, Louisiana, August 1995.

11. Voyiadjis, G. Z., and Thiagarajan, G., "A Cyclic Plasticity Model for Metal Matrix Composites." Invited lecture presented at the Symposium on "Recent Advances in Constitutive Laws for Engineering Materials," at the International Conference on Computational Engineering Science (ICES-95), Mauna Lani, Hawaii, August 1995.

12. Voyiadjis, G. Z., and Park, T., "Finite Element Analysis of Damage and Elasto-Plastic Behavior of Metal Matrix Composites." Invited lecture presented at the Symposium on "Inelastic Deformation, Damage and Life Analysis," at the International Conference on Computational Engineering Science (ICES-95), Mauna Lani, Hawaii, August 1995.

13. Voyiadjis, G. Z., and Thiagarajan, G., "A Damage Cyclic Plasticity Model for Metal Matrix Composites." Invited lecture presented at the Symposium on "Micromechanics and Constitutive Modelling of Composite Materials," at the ASME joint Applied Mechanics and Materials Summer Conference AMD-MD '95, Los Angeles, California, June 1995.

14. Voyiadjis, G. Z., and Park, T., "Finite Element Analysis of Interfacial and Local Damage in Metal Matrix Composites." Invited lecture presented at the Symposium on "Recent Advances in Damage Mechanics," at the ASME joint Applied Mechanics and Materials Summer Conference AMD-MD '95, Los Angeles, California, June 1995.

15. Voyiadjis, G. Z., Park, T., and Venson, A. R., "Finite Element Implementation of the Overall Approach to Damage in Metal Matrix Composites." Invited lecture presented at the Symposium on "Damage Mechanics in Engineering Materials," at the 10th ASCE Engineering Mechanics Conference, Boulder, Colorado, May 1995.

16. Voyiadjis, G. Z., "Damage in Materials." Invited lecture presented at Louisiana State University, Saturday Science at LSU Seminar Series, Baton Rouge, Louisiana, December 1994.

17. Voyiadjis, G. Z., and Park, T., "Interfacial and Local Damage Analysis of Metal Matrix Composites." Invited lecture presented at the Symposium on "Damage Mechanics in Composites," at the ME '94, The International Mechanical Engineering Congress and Exposition, Chicago, Illinois, November 1994.

18. Voyiadjis, G. Z., "Constitutive Modeling of Plasticity with Damage in Metal Matrix Composites." Invited lecture presented at the National Science Foundation Workshop on Mechanics and Processing of Advanced Engineering Materials - II, Atlanta, Georgia, October, 1994.

19. Voyiadjis, G. Z., Venson, A.R., and Kattan, P.I. "Damage Approaches to Unidirectional Elastoplastic Laminas Under Uniaxial Tension." Invited lecture presented at the 31st

Annual Technical Meeting of the Society of Engineering Science, College Station, Texas, October 1994.

20. Voyiadjis, G. Z., and Kattan, P. I., "Damage and Plastic Deformation in a Uniaxially Loaded Thin Composite Ply." Invited lecture presented at the First International Conference on Composites Engineering, New Orleans, Louisiana, August 1994.

21. Voyiadjis, G. Z., and Kattan, P. I., "Micromechanical Modeling of Damage and Plasticity in Continuously Reinforced Metal Matrix Composites." Invited lecture presented at the 12th U.S. National Congress of Applied Mechanics, Seattle, Washington, June 1994.

22. Voyiadjis, G. Z., "Characterization of Damage in Metal Matrix Composites." Invited lecture presented in the Department of Aerospace Engineering, Mechanics and Engineering Science, University of Florida, Gainesville, Florida, March 1994.

23. Voyiadjis, G. Z., "Damage Accumulation in Metal Matrix Composites." Invited lecture presented at the National Science Foundation Workshop on "Processing and Constitutive Modelling of Advanced Engineered Materials - I," Washington State University, Pullman, Washington, October 1993.

24. Voyiadjis, G. Z., "Damage in Metal Matrix Composites." Invited lecture presented in the Department of Mechanical Engineering, University of Delaware, Newark, Delaware, October 1993.

25. Voyiadjis, G. Z., Venson, A. R., and Kattan, P. I., "Damage Mechanism in Metal Matrix Composite Plates." Invited lecture presented at the Army Research Office Workshop on Dynamic Response of Composite Structures, New Orleans, Louisiana, August 1993.

26. Voyiadjis, G. Z., Kattan, P. I., and Venson, A. R., "Evaluation of a Damage Tensor for Metal Matrix Composites." Presented at the MECAMAT '93, International Seminar on Micromechanics of Materials, Fontainebleau, France, July 1993.

27. Voyiadjis, G. Z., and Huang, W., "Microstructural Characterization of Continuum Plasticity Theory with Finite Strains." Invited lecture presented at the Fourth International Symposium on Plasticity and Its Current Applications, Baltimore, Maryland, July 1993.

28. Voyiadjis, G. Z., and Kattan, P. I., "Damage and Inelastic Deformation of SiC/Ti-Aluminide Composites." Invited lecture presented at the MEET'N'93, First SES-ASME-ASCE Joint Meeting on Mechanics, Symposium on Inelastic Micromechanics in SiC/Ti Composites, Charlottesville, VA, June 1993.

SESSIONS ORGANIZED BY DR. VOYIADJIS

1. Co-Organizer and Co-Chairman of four sessions of the Symposium on "Damage and Interfacial Debonding in Composites," held in the 32nd Society of Engineering Science Meeting, New Orleans, Louisiana, October/November 1995.

2. Co-Organizer and Co-Chairman of four sessions of the Symposium on "Micromechanics of Fibers or Composites," held in the 2nd International Conference on Composites Engineering, New Orleans, Louisiana, August 1995.

3. Co-Organizer and Co-Chairman of four sessions of the Symposium on "Fatigue of Composites," held in the 2nd International Conference on Composites Engineering, New Orleans, Louisiana, August 1995.

4. Co-Organizer and Co-Chairman of five sessions of the Symposium on "Inelasticity and Micromechanics of Metal Matrix Composites," held in the Twelfth U.S. National Congress of Applied Mechanics, Seattle, Washington, June/July 1994.

5. Panel Member of the session on "Damage Accumulation and Life Prediction in Metal Matrix Composites," of the National Science Foundation Workshop on "Processing and Constitutive Modeling of Advanced Engineered Materials," Washington State University, Pullman, Washington, October 1993.

6. Organizer and Co-Chairman of four sessions on "Modeling and Applications in Microstructural Characterization of Large Plastic Deformations," held in the Fourth International Symposium on Plasticity and its Current Applications, Baltimore, Maryland, July 1993.

7. Co-Organizer and Co-Chairman of four sessions of the Maciej P. Bieniek Symposium on "Mechanics of Materials and Structures," held in the MEET'N'93, First Joint ASCE-ASME-SES Mechanics Conference, University of Virginia, Charlottesville, V irginia, June 1993.

POST DOCTORAL ASSOCIATES SUPERVISED

1. Dr. Taehyo Park, Ph.D. in Civil Engineering (Damage Mechanics), December 1994, Louisiana State University, Term of appointment: 2/15/95 -present.

2. Dr. Anthony R. Venson, Ph.D. in Civil Engineering (Damage in Metal Matrix Composites), December 1994, Louisiana State University, Term of appointment: 1/1/95 - 7/31/95. Currently: Visiting Assistant Professor, Department of Civil Engineering, University of Southwestern Louisiana, Lafayette, Louisiana.

STUDENTS SUPERVISED

(a) Ph.D. Students

1. Ganesh Thiagarajan: A Cyclic Plasticity/Damage Model for Metal Matrix Composites; May 1996, LSU. Currently: Instructor, Department of Civil and Environmental Engineering, Louisiana State University, Baton Rouge, Louisiana.

2. A. R. Venson: Experimental Macro and Microstructural Characterization of Damage for Metal Matrix Composites; December 1994, LSU. Currently: Visiting Assistant Professor, Department of Civil Engineering, University of Southwestern Louisiana, Lafayette, Louisiana.

3. Taehyo Park: Finite Element Analysis of Damage and Elastic-Plastic Behavior of Metal Matrix Composites; December 1994, LSU. Currently: Research Associate, Department of Civil and Environmental Engineering, Louisiana State University, Baton Rouge, Louisiana.

(b) M.S. Students

1. Babur Deliktas: Damage in Metal Matrix Composite Using the Generalized Cells Model; May 1996, LSU.

Chapter 1

A Coupled Incremental Damage and Plasticity Theory for Metal Matrix Composites

1.1 Introduction

A coupled incremental damage and plasticity theory for metal matrix composites is introduced here. This coupling occurs only in the matrix since the fibers are assumed to be only elastic. This allows damage to be path dependent either on the stress history or the thermodynamic force conjugate to damage. This is achieved through the use of incremental damage tensors. Damage and plastic deformations are incorporated in the proposed model that is used for the analysis of fiber-reinforced metal matrix composite materials. The proposed micro-mechanical damage relations are used for each of the matrix and the fiber. This is coupled with the interfacial damage between the matrix and the fiber exclusively. The damage relations are linked to the overall response through a certain homogenization procedure. Two local incremental damage tensors $\boldsymbol{m}^m$ and $\boldsymbol{m}^f$ are used where $\boldsymbol{m}^m$ accounts for the damage in the ductile matrix such as nucleation and growth of voids, while $\boldsymbol{m}^f$ reflects the incremental damage in the fibers such as fracture. An additional incremental tensor $\boldsymbol{m}^d$ is incorporated in the overall formulation that represents interfacial damage between the matrix and the fiber. An overall incremental damage tensor, $\boldsymbol{m}$, is introduced that accounts for all these separate damage tensors $\boldsymbol{m}^m$, $\boldsymbol{m}^f$ and $\boldsymbol{m}^d$.

For the undamaged matrix material, a von Mises type yield criterion with an associated flow rule, and Ziegler-Prager kinematic hardening rule are used. However, the resulting overall yield function for the damaged composite is anisotropic. The overall kinematic hardening rule for the damaged composite system is a combination of the generalized Ziegler-Prager rule and the Phillips-type rule . The elasto-plastic stiffness tensor is derived for the damaged composite.

Evaluation of the material parameters is done for the case of isotropic damage.

1.2 Formulation of the Incremental Damage Tensor

1.2.1 Total Damage Tensor M

In order to obtain the incremental damage tensor m the concept of effective stress as first used by Kachanov (1958)[1] is presented for the one dimensional stress state. The incremental relations are subsequently derived using Kachanov's concept. C_0 in this work refers to the initial undeformed and undamaged configuration of the body and C is the corresponding configuration of the body that is both deformed and damaged after a set of external agencies act on it. The state of the body after it has only deformed without damage (by removing the damage fictitiously) is denoted by $\bar{C}$. This is presented by Voyiadjis and Kattan (1992a)[2].

A linear transformation between the Cauchy stress in the configuration C, and the effective Cauchy stress in the configuration $\bar{C}$ is assumed such that

$$\bar{\sigma}\bar{A} \;=\; \sigma A \tag{1.1}$$

or

$$\bar{\sigma} \;=\; (1 - \phi)^{-1}\sigma \tag{1.2}$$

where

$$\phi \;=\; \frac{A - \bar{A}}{A} \tag{1.3}$$

In the above equations A and $\bar{A}$ are the areas of crossections of the axially loaded bar in the C and $\bar{C}$ configurations respectively. ϕ is a measure of damage. Making use of equation (1.2) an incremental formulation for damage is obtained such that

$$\dot{\bar{\sigma}} \;=\; (1 - \phi)^{-1}\dot{\sigma} + (1 - \phi)^{-2}\sigma\dot{\phi} \tag{1.4}$$

The concept of effective stress as generalized by Murakami (1988)[3] is given through the generalization of equation (1.2) such that

$$\bar{\sigma} \;=\; M : \sigma \tag{1.5}$$

or

$$\bar{\sigma}^r = M^r : \sigma^r \qquad r = m, f \tag{1.6}$$

for the individual constituents of the matrix and the fiber respectively. where M is the fourth-order damage effect tensor and is a function of the second order symmetric tensor ϕ.

The effective Cauchy stress tensor, $\bar{\sigma}$, need not be symmetric or frame invariant. Once the effective Cauchy stress, $\bar{\sigma}$, is symmetrized, it can be shown that it satisfies the frame invariance principle (Voyiadjis and Kattan, 1992a)[2].

The fourth order tensor M can be represented by a 6X6 matrix as a function of $(I_2 - \phi)$ in the form (Murakami)[3]

$$[M] = [M(I_2 - \phi)] \tag{1.7}$$

2

where I_2 is the second-order identity tensor. Murakami [3] has shown that ϕ is symmetric which is the generalization of the scalar variable ϕ. The stress tensor $\bar{\sigma}$, in conjunction with the matrix form of M given by equation (1.7), is represented by a vector given by

$$[\bar{\sigma}] = [\bar{\sigma}_{11}, \bar{\sigma}_{22}, \bar{\sigma}_{33}, \bar{\sigma}_{12}, \bar{\sigma}_{23}, \bar{\sigma}_{31}]^T \tag{1.8}$$

The symmetrized effective Cauchy stress tensor $\bar{\sigma}$ used here is expressed by (Lee, et al., 1986)[4]

$$\bar{\sigma}_{ij} = \frac{1}{2}[\sigma_{ik}(\delta_{kj} - \phi_{kj})^{-1} + (\delta_{il} - \phi_{il})^{-1}\sigma_{lj}] \tag{1.9}$$

which is a second rank tensorial generalization of the scalar equation (1.2)

The stress given by equation (1.9) is frame-independent. Utilizing the symmetrization procedure outlined by equation (1.9) the corresponding 6X6 matrix form of tensor M is given by Voyiadjis and Kattan (1992a)[2] as follows:

$$[M] = \frac{1}{2\nabla} \begin{bmatrix} 2\omega_{22}\omega_{33} - 2\phi_{23}^2 & 0 & 0 \\ 0 & 2\omega_{11}\omega_{33} - 2\phi_{13}^2 & 0 \\ 0 & 0 & 2\omega_{11}\omega_{22} - 2\phi_{12}^2 \\ \phi_{13}\phi_{23} + \phi_{12}\omega_{33} & \phi_{13}\phi_{23} + \phi_{12}\omega_{33} & 0 \\ 0 & \phi_{12}\phi_{13} + \phi_{23}\omega_{11} & \phi_{12}\phi_{13} + \phi_{23}\omega_{11} \\ \phi_{12}\phi_{23} + \phi_{13}\omega_{22} & 0 & \phi_{12}\phi_{23} + \phi_{13}\omega_{22} \end{bmatrix} \tag{1.10}$$

$$\begin{bmatrix} 2\phi_{13}\phi_{23} + 2\phi_{12}\omega_{33} & 0 & 2\phi_{12}\phi_{23} + 2\phi_{13}\omega_{22} \\ 2\phi_{13}\phi_{23} + 2\phi_{12}\omega_{33} & 2\phi_{12}\phi_{13} + 2\phi_{23}\omega_{11} & 0 \\ 0 & 2\phi_{12}\phi_{13} + 2\phi_{23}\omega_{22} & \phi_{12}\phi_{13} + \phi_{13}\omega_{11} \\ \omega_{22}\omega_{33} + \omega_{11}\omega_{33} - \phi_{23}^2 - \phi_{13}^2 & \phi_{12}\phi_{23} + \phi_{13}\omega_{22} & \phi_{12}\phi_{23} + \phi_{23}\omega_{11} \\ \phi_{12}\phi_{23} + \phi_{13}\omega_{22} & \omega_{11}\omega_{33} + \omega_{11}\omega_{22} - \phi_{13}^2 - \phi_{12}^2 & \phi_{13}\phi_{23} + \phi_{12}\omega_{33} \\ \phi_{12}\phi_{13} + \phi_{23}\omega_{11} & \phi_{13}\phi_{23} + \phi_{12}\omega_{33} & \omega_{22}\omega_{33} + \omega_{11}\omega_{22} - \phi_{23}^2 - \phi_{12}^2 \end{bmatrix}$$

where ∇ Is given as follows

$$\nabla = \omega_{11}\omega_{22}\omega_{33} - \phi_{23}^2\omega_{11} - \phi_{13}^2\omega_{22} - \phi_{12}^2\omega_{33} - 2\phi_{12}\phi_{23}\phi_{13} \tag{1.11}$$

The notation ω_{ij} is used here to denote $\delta_{ij} - \phi_{ij}$. The physical characterization of damage, ϕ is presented in section 4.

1.2.2   Incremental Damage Tensor m

The incremental relation of equation (1.2) is given by the following expression

$$d\bar{\sigma} = (1 - \phi)^{-1}d\sigma + (1 - \phi)^{-2}d\phi\,\sigma \tag{1.12}$$

3

Equation (1.12) may also be obtained using Figure 1.1. In the local sub-configurations of the matrix, C^m, and fiber, C^f, equation (1.12) isexpressed as follows

$$d\bar{\sigma}^r = (1 - \phi^r)^{-1} d\sigma^r + (1 - \phi^r)^{-2} d\phi^r \, \sigma^r \tag{1.13}$$

for $r = m, f$. The generalization of the concept of the incremental relation given by equation (1.12) is obtained by introducing the incremental relation of equation (1.6) such that

$$d\bar{\sigma}^r = M^r : d\sigma^r + dM^r : \sigma^r \tag{1.14}$$

or

$$\dot{\bar{\sigma}}^r = M^r : \dot{\sigma}^r + \dot{M}^r : \sigma^r \tag{1.15}$$

The superposed dot indicates material time differentiation. In order for equation (1.15) to be homogeneous in time of order one (i.e. stress-rate independent) $\dot{M}^r$ should be a linear function of $\dot{\sigma}^r$. It will be demonstrated in this work that the following relation exists

$$\dot{\phi}^r = X^r : \dot{\sigma}^r \tag{1.16}$$

Since M^r is a function of ϕ^r, therefore

$$\dot{M}^r_{ijkl} = \frac{\partial M^r_{ijkl}}{\partial \phi^r_{pq}} \dot{\phi}^r_{pq} \tag{1.17}$$

Consequently the resulting relation between $\dot{M}^r$ and $\dot{\sigma}^r$ Is such that

$$\dot{M}^r_{ijkl} = \frac{\partial M^r_{ijkl}}{\partial \phi^r_{pq}} X^r_{pqrs} \dot{\sigma}^r_{pq} \tag{1.18}$$

or

$$\dot{M}^r = \Gamma^r : \dot{\sigma}^r \tag{1.19}$$

where Γ^r is a sixth order tensor.

Equation (1.19) may also be written in terms of the strain rate by making use of the fourth order elasto-plastic stiffness tensor D^r, such that

$$\dot{M}^r = \Gamma^r : D^r : \dot{\epsilon}^r \tag{1.20}$$

Making use of equation (1.19) in equation (1.15) one obtains the incremental damage expression such that

$$\dot{\bar{\sigma}}^r = m^r : \dot{\sigma}^r \tag{1.21}$$

where

$$m^r_{ijkl} = M^r_{ijkl} + \Gamma^r_{ijpqkl} \sigma^r_{pq} \tag{1.22}$$

The fourth order tensor m^r could be interpreted as the incremental damage tensor as opposed to the total damage tensor M^r. During elastic loading and unloading, equation (1.21) could be substituted by equation (1.5).

4

1.3 Relation Between the Cumulative Damage and the Local Damage Tensors

1.3.1 Basic Assumptions

The metal matrix composite system used in this work is restricted to small deformations with infinitesimal strains. The material consists of an elasto-plastic ductile metal matrix reinforced by elastic aligned continuous fibers. C_0 denotes the initial undeformed and undamaged configuration of a single lamina while C_0^m and C_0^f are the initial matrix and fiber sub-configurations for the single lamina respectively. The composite material is assumed to undergo elasto-plastic deformation and damage due to the applied loads. The corresponding resulting overall configuration for a single lamina is denoted by C while the respective matrix and fiber local sub-configurations for a single lamina are denoted by C^m and C^f respectively. Damage is expressed by generalizing the concept proposed by Kachanov (1958) [1] whereby two kinds of fictitious configurations $\bar{C}$ and $\tilde{C}$ of the composite system at the lamina level are considered as shown in Figure 1.2. Configuration $\bar{C}$ is obtained from C by removing the different types of damages that the single lamina has undergone due to the applied stresses. However, configuration $\tilde{C}$ is obtained from C for a single lamina by removing only the interfacial damage between the matrix and the fiber. The total or incremental stress at configuration C is converted to the respective total or incremental stress at the fictitious configuration $\bar{C}$ through the damage tensors $\boldsymbol{m}$ or $\boldsymbol{M}$ respectively as indicated in Figure 1.2. Configuration $\bar{C}$ is termed full effective configuration, while $\tilde{C}$ is the partial effective configuration.

The coupled formulation of the plastic flow and damage propagation seems to be impossible, due to the presence of the two different dissipative mechanisms that influence each other. This could be indicated by the fact that the position of the slip planes affects the orientation of nucleated micro-cracks. However, one can assume that the energy dissipated in the yielding and damaging processes be independent of each other and apply a phenomenological model of interaction. In this work use is made of the concept of effective stress (Lemaitre, 1971)[5]. Making use of a fictitious undamaged system, the dissipation energy due to plastic flow in this undamaged system is assumed to be equal to the dissipation energy due to plastic flow in the real damaged system.

The main feature of the present approach is that local effects of damages are considered at both the single lamina level as well as the laminate level. The damages at the single lamina level are described separately by the damage in the matrix, damage in the fiber, and interfacial damage between the matrix and the fiber. This is schematically indicated in Figure 1.2 where the undamaged matrix and fiber configurations $\bar{C}^m$ and $\bar{C}^f$, respectively, are transformed to their respective damaged configurations $\tilde{C}^m$ and $\tilde{C}^f$ through the incremental fourth order damage tensors $\boldsymbol{m}^m$ and $\boldsymbol{m}^f$. $\boldsymbol{m}^m$ reflects damage in the matrix only and accordingly $\boldsymbol{m}^f$ reflects damage in fibers only. These configurations could also be transformed through the fourth order damage tensors $\boldsymbol{M}^m$ and $\boldsymbol{M}^f$ as shown in Figure 1.2. However, as will be demonstrated later the local incremental damage tensors are better suited for use in the formulation of the constitutive equation of the damaged material behavior. One can also easily express the overall incremental damage tensor $\boldsymbol{m}$ in terms of its local components

$\boldsymbol{m}^m$, $\boldsymbol{m}^f$ and $\boldsymbol{m}^d$. Furthermore, equations are considerably simplified and it also yields a more efficient computational solution for the boundary value problems.

Damage tensors $\boldsymbol{m}^m$ and $\boldsymbol{M}^m$ reflect damage in matrix only and accordingly $\boldsymbol{m}^f$ and $\boldsymbol{M}^f$ reflect damage in fibers only. Following this local damage description, the local-overall relations are used to transfer the local damage effects to the whole composite system in configuration $\tilde{C}$ as shown in Figure 1.2. This is accomplished through the stress concentration tensors $\boldsymbol{B}^m$ and $\boldsymbol{B}^f$ of the matrix and fibers, respectively.

The effect of interfacial damage between the fibers and the matrix is represented by a serial incremental transformation $\boldsymbol{m}^d$ (or equivalently $\boldsymbol{M}^d$) and transforms the configuration $\tilde{C}$ to the final damaged configuration C. Referring to Figure 1.2, the local nature of damage of this approach for the single lamina is clear, and different damages are separately isolated.

Referring to Figure 1.2 this approach is summarized in the following three steps. The incremental local damage tensors $\boldsymbol{m}^m$ and $\boldsymbol{m}^f$ are first applied to the local effective configurations $\bar{C}^m$ and $\bar{C}^f$, respectively. This is followed by applying the damage stress concentration factors $\boldsymbol{B}^m$ and $\boldsymbol{B}^f$ to the local partial effective configurations $\tilde{C}^m$ and $\tilde{C}^f$ in order to obtain the overall partial effective configuration $\tilde{C}$. Finally, the incremental interfacial damage tensor $\boldsymbol{m}^d$ is applied to the overall partial effective configuration $\tilde{C}$ in order to obtain the overall damaged configuration C of a single lamina.

The tensor $\boldsymbol{m}^m$ encompasses all the pertinent damage related to the matrix while the tensor $\boldsymbol{m}^f$ reflects the damage pertinent to the fibers (Voyiadjis and Kattan, 1993)[6]. However, the interfacial damage tensor $\boldsymbol{M}^d$ is related to the interfacial damage variable ϕ^d. An interfacial damage variable can be defined through the use of an RVE (Representative Volume Element) as indicated by Voyiadjis and Park (1995)[7]

$$\phi^d = \frac{S - \bar{S}}{S} \tag{1.23}$$

where S is the total interfacial length, between the fiber and the matrix and $\bar{S}$ is the effective (net) resisting length corresponding to the total interfacial length in contact.

1.3.2   Theoretical Formulation of $\boldsymbol{m}$

An incremental overall damage tensor $\boldsymbol{m}$ is introduced for the whole composite system as shown in Figure 1.1. This tensor is defined similarly to the definitions of $\boldsymbol{m}^m$, $\boldsymbol{m}^f$ and $\boldsymbol{m}^d$ such that

$$\dot{\bar{\boldsymbol{\sigma}}} = \boldsymbol{m} : \dot{\boldsymbol{\sigma}} \tag{1.24}$$

or

$$d\boldsymbol{\sigma} = \boldsymbol{m} : d\boldsymbol{\sigma} \tag{1.25}$$

$\boldsymbol{m}$ reflects all types of damages that the composite undergoes including the damage due to the interaction between the matrix and fibers. Similarly a tensor $\boldsymbol{M}$ is used for the total stresses (Voyiadjis and Kattan, 1993 [6], Kattan and Voyiadjis (1993b) [8]) as indicated by equation (1.5). The matrix representation for $\boldsymbol{M}$ was explicitly derived by expressing the

stress in vector form. The tensor M as well as tensor $\dot{M}$ can be shown to be symmetric. It follows from equation (1.22) that the incremental damage tensor m is also symmetric. This property will be used extensively in the derivation that follows. The same holds true for tensors m^m, m^f and m^d.

Similar to tensor M^d, both tensors M^m and M^f could be represented in terms of second order tensors ϕ^m and ϕ^f respectively as shown in Voyiadjis and Kattan (1993)[6] and Kattan and Voyiadjis (1993b)[8]. It therefore follows that m^d, m^m and m^f, could be represented in terms of tensors $\dot{\phi}^d$, $\dot{\phi}^m$ and $\dot{\phi}^f$ respectively, ϕ^d, ϕ^m and ϕ^f respectively and σ. The overall effective Cauchy stress $\bar{\sigma}$ is related to the local effective Cauchy stress $\bar{\sigma}^m$ and $\bar{\sigma}^f$ by making use of the micro-mechanical model proposed by Dvorak and Bahei-El-Din (1979)[9] such that

$$\dot{\bar{\sigma}} = \bar{c}^m \dot{\bar{\sigma}}^m + \bar{c}^f \dot{\bar{\sigma}}^f \qquad (1.26)$$

where $\bar{c}^m$ and $\bar{c}^f$ are the effective matrix and fiber volume fractions, respectively. The effective matrix Cauchy stress rate and the corresponding fiber Cauchy stress rate are defined as follows:

$$\dot{\bar{\sigma}}^m = m^m : \dot{\tilde{\sigma}}^m \qquad (1.27)$$

and

$$\dot{\bar{\sigma}}^f = m^f : \dot{\tilde{\sigma}}^f \qquad (1.28)$$

where $\dot{\tilde{\sigma}}^m$ and $\dot{\tilde{\sigma}}^f$ are the partial effective stress rates in the $\tilde{C}^m$ and $\tilde{C}^f$ configurations, respectively. These stresses are termed partial effective since the interfacial damage has not yet been incorporated into the formulation. Referring to Figure 1.2 and making use of the partial stress concentrations $\tilde{B}^m$ and $\tilde{B}^f$, the corresponding partial effective matrix Cauchy stress rate and corresponding fiber Cauchy stress rate are given by the following equations:

$$\dot{\tilde{\sigma}}^m = \tilde{B}^m : \dot{\tilde{\sigma}} \qquad (1.29)$$

and

$$\dot{\tilde{\sigma}}^f = \tilde{B}^f : \dot{\tilde{\sigma}} \qquad (1.30)$$

The partial effective overall composite Cauchy stress rate $\dot{\tilde{\sigma}}$ is defined as

$$\dot{\tilde{\sigma}} = m^d : \dot{\sigma} \qquad (1.31)$$

in terms of the Cauchy stress rate $\dot{\sigma}$. One can also define the incremental fourth order damage tensor $\tilde{m}$ such that

$$\dot{\bar{\sigma}} = \tilde{m} : \dot{\tilde{\sigma}} \qquad (1.32)$$

Making use of relations (1.27) and (1.28) in equation (1.26) one obtains the following expression:

$$\dot{\bar{\sigma}} = \bar{c}^m m^m : \dot{\tilde{\sigma}}^m + \bar{c}^f m^f : \dot{\tilde{\sigma}}^f \qquad (1.33)$$

Substituting into equation (1.33) for the partial effective matrix and fiber stress rates from relations (1.29) and (1.30) respectively and making use of equation (1.31) the resulting equation is given as follows:

$$\dot{\bar{\sigma}} = (\bar{c}^m \boldsymbol{m}^m : \tilde{\boldsymbol{B}}^m + \bar{c}^f \boldsymbol{m}^f : \tilde{\boldsymbol{B}}^f) : \boldsymbol{m}^d : \dot{\boldsymbol{\sigma}} \qquad (1.34)$$

Comparing equation (1.34) with expression (1.21) one concludes that the following relation is obtained between the overall incremental damage tensor $\boldsymbol{m}$ and its components $\boldsymbol{m}^m$, $\boldsymbol{m}^f$ and $\boldsymbol{m}^d$:

$$\boldsymbol{m} = (\bar{c}^m \boldsymbol{m}^m : \tilde{\boldsymbol{B}}^m + \bar{c}^f \boldsymbol{m}^f : \tilde{\boldsymbol{B}}^f) : \boldsymbol{m}^d \qquad (1.35)$$

This expression defines the cumulative incremental damage of the composite as a function of its local components.

1.4   Experimental Determination of Damage

Damage is characterized here as micro-cracks. However, any other type of damage could be included in this formulation. The physical interpretation of the damage tensor ϕ^r is also presented here. The tensor ϕ^r is evaluated experimentally for two different types of laminate layups. For each case ϕ^m and ϕ^f is computed independently.

	Matrix (Ti-14Al-21Nb)	Fiber (SiC)
Modulus	$8 \times 10^4\,MPa$	$41 \times 10^4\,MPa$
Poisson's Ratio	0.30	0.22
Initial Volume Fraction	0.65	0.35
Yielding Stress $\bar{\sigma}_0^m$	$360\,MPa$	
Kinematic hardening Parameter b	$90\,MPa$	

Table 1.1: Material Properties

Voyiadjis and Venson (1995)[10] presented experimental investigations and procedures for the determination of damage for the macro- and the micro-analysis of a SiC-Titanium Aluminide metal matrix composite with material properties indicated in Table 1.1. Uniaxial tension tests are performed in the work on laminate specimens of two different layups. Flat dogbone shaped specimens are fabricated from each of the layups. For each of the different layups, specimens are loaded to various load levels ranging from 70% of the load rupture to the rupture load at room temperature. Making use of this experimental procedure, damage evolution is experimentally evaluated through a quantitative micro-analysis technique. This analysis is performed using scanning electron microscopy on three mutually perpendicular representative cross-sections from a representative volume element defined for the theoretical development of damage evolution.

The damage tensor proposed by Voyiadjis and Venson(1995)[10] is defined for a general state of loading based upon the experimental observations of crack densities on tree mutually

8

perpendicular cross-sections of the specimens. The damage tensors, ϕ^f, ϕ^m and ϕ^d are defined as a second-rank tensor in the form of

$$\phi^r = \bar{\rho}^r \otimes \bar{\rho}^r \qquad (1.36)$$

and in the matrix form as follows

$$[\phi^r] = \begin{bmatrix} \bar{\rho}_x\bar{\rho}_x & \bar{\rho}_x\bar{\rho}_y & \bar{\rho}_x\bar{\rho}_z \\ \bar{\rho}_y\bar{\rho}_x & \bar{\rho}_y\bar{\rho}_y & \bar{\rho}_y\bar{\rho}_z \\ \bar{\rho}_z\bar{\rho}_x & \bar{\rho}_z\bar{\rho}_y & \bar{\rho}_z\bar{\rho}_z \end{bmatrix} \qquad (1.37)$$

Where ρ_i for $i = x, y, z$ is the normalized crack density on a cross-section whose normal is along the i-axis. The crack density on the representative volume element for the ith cross-section is calculated as follows:

$$\bar{\rho}_i = \frac{\rho_i}{m\rho^*} \qquad (1.38)$$

and

$$\rho_i = \frac{l_i}{A_i} \qquad (1.39)$$

In the above expressions, l_i is the total length of the cracks on the ith cross-sectional area for each constituents, m is a normalization factor chosen so that the values of the damage variable ϕ^r fall within the expected range $0 \leq \phi^r_{ij} < 1$, and ρ^* is defined as follows

$$\rho^* = \sqrt{\rho^2_{x_{max}} + \rho^2_{y_{max}} + \rho^2_{z_{max}}} \qquad (1.40)$$

Where $\rho_{x_{max}}$, $\rho_{y_{max}}$ and $\rho_{z_{max}}$ are the values of l_i/A_i at the maximum load (rupture) on the respective cross-section.

The damage tensor obtained experimentally from equation (1.36) is used in the constitutive equations in order to predict the mechanical behavior of the composite system. This procedure may be used independently to quantify each of the damages in the matrix and the fiber. Use is made of the scanning electron microscope in order to quantify the damage tensor ϕ^r expressed by equations (1.36) and (1.37). This is performed at various load levels ranging from rupture load down to 70 % of the rupture at room temperature (Voyiadjis and Venson, 1995)[10]. A more extensive investigation would be to start from the onset of load all the way to rupture. However, the damage tensor ϕ^r determined experimentally by Voyiadjis and Venson (1995)[10] is between 70 % and of the rupture load to the final rupture for two types of laminate layups $(0/90)_s$ and $(\pm 45)_s$ each consisting of four plies. These layups are examined both numerically and experimentally (Voyiadjis and Venson, 1995[10]). Since in order to quantify damage, the specimen has to be sectioned, consequently different specimens are used to quantify damage at various percentages of the rupture load.

The crack densities ρ_i measured experimentally by the procedure outlined above are shown in Tables 1.2 and 1.3 for the $(0/90)_s$ and $(\pm 45)_s$ layups, respectively. The normalized values $\bar{\rho}_i$ are calculated using the values of ρ_i obtained previously for each layups. The results

are then used to calculate the damage variable ϕ^r based on expression (1.37). Damage strain curves are consequently generated for each layups orientation. These damage values can then be used in the constitutive model to accurately predict the mechanical behavior of metal matrix composites. For a more complete discussion on the physical characterization the reader is referred to a paper by Voyiadjis and Venson (1995)[10].

%Load	%Strain	$\rho_x^m \times 10^{-4}$ mm/mm^2	$\rho_x^f \times 10^{-4}$ mm/mm^2	$\rho_y^m \times 10^{-4}$ mm/mm^2	$\rho_y^f \times 10^{-4}$ mm/mm^2
70	.3182	0.00	41.82	0.00	3.41
75	.4487	0.00	70.32	0.00	36.40
80	.4611	0.00	100.77	—	—
85	.5202	0.00	106.24	0.00	56.43
90	.5808	0.00	126.68	.77	66.94

Table 1.2: Local Crack densities for $(0/90)_S$ laminate

%Load	%Strain	$\rho_x^m \times 10^{-4}$	$\rho_x^f \times 10^{-4}$	$\rho_y^m \times 10^{-4}$	$\rho_y^f \times 10^{-4}$
70	.2414	0.00	49.23	—	—
75	.2779	0.00	49.32	0.00	42.44
80	.4324	0.00	51.84	0.00	101.29
85	.5268	0.00	52.99	0.00	117.01
90	.5729	0.00	56.67	48.98	97.61

Table 1.3: Local Crack densities for $(\pm 45)_S$ laminate

1.5 Damage Criterion

In this work it is assumed that matrix undergoes ductile damage while the fiber undergoes brittle damage.The mechanisms of interfacial damage are dependent on the fiber direction. It is clear that the damage mechanism for each of the constituents of the composite materials is different from the other and therefore one single damage micro-mechanism can not be considered for the three types of damages outlined here for a single lamina. Each type of damage evolution is considered separately.

In order to obtain a damage criterion for non-proportional loading, the anisotropy of damage increase (hardening) must be considered. The anisotropic damage criterion used here is expressed in terms of a tensorial hardening parameter, $\boldsymbol{h}$ (Voyiadjis and Park, 1995)[7].

$$g^r \equiv g^r(\boldsymbol{Y}^r, \kappa^r) = 0 \quad , \quad r = m, f, d \tag{1.41}$$

or

$$g^r \equiv P_{ijkl}^r(Y_{ij}^r - \gamma_{ij}^r)(Y_{kl}^r - \gamma_{kl}^r) - 1 = 0 \tag{1.42}$$

The generalized thermodynamic backsterss force γ_{ij}^r, which is a concequence of crack interactions is given by

$$\gamma_{ij}^r = c^r \dot{\phi}_{ij}^r \qquad (1.43)$$

where c^r is a material constant and

$$P_{ijkl}^r = h_{ij}^{-r} h_{kl}^{-r} \qquad (1.44)$$

Here $\boldsymbol{Y}^r$ is a generalized thermodynamic force conjugate to the damage tensor $\boldsymbol{\phi}^r$ for each of the damage associated with the matrix, fiber and debonding. The hardening tensor $\boldsymbol{h}^r$ is expressed as

$$h_{ij}^r = u_{ij}^r + V_{ij}^r \qquad (1.45)$$

Tensors $\boldsymbol{u}^r$ and $\boldsymbol{V}^r$ (which represents the initial threshold against damage for the material) are defined for orthotropic materials(Voyiadjis and Park, 1995)[7] in terms of the generalized Lamé constants $\lambda_1^r, \lambda_2^r, \lambda_3^r$ and v_1^r, v_2^r, v_3^r as follows

$$\boldsymbol{u} = \begin{pmatrix} \lambda_1^r \eta_1^r \left(\frac{\kappa^r}{\lambda_1^r} \right)^{\xi_1^r} & 0 & 0 \\ 0 & \lambda_2^r \eta_2^r \left(\frac{\kappa^r}{\lambda_2^r} \right)^{\xi_2^r} & 0 \\ 0 & 0 & \lambda_3^r \eta_3^r \left(\frac{\kappa^r}{\lambda_3^r} \right)^{\xi_3^r} \end{pmatrix} \qquad (1.46)$$

and

$$\boldsymbol{V}^r = \begin{pmatrix} \lambda_1^r v_1^{r2} & 0 & 0 \\ 0 & \lambda_2^r v_2^{r2} & 0 \\ 0 & 0 & \lambda_3^r v_3^{r2} \end{pmatrix} \qquad (1.47)$$

where κ^r is a scalar representing the total damage energy and given by

$$\kappa^r = \int_0^t \boldsymbol{Y}^r : \dot{\boldsymbol{\phi}}^r dt \quad , \quad r = m, f, d \qquad (1.48)$$

For a simpler damage model η_i^r may be eliminated by setting it equal to unity and seting $\xi_1^r = \xi_2^r = \xi_3^r$.This causes less flexibility in the behavior of the damage. However, this allows the reduction of material parameters by five.

The generalized Lamé constants in this work are defined as follows

$$\lambda_i^r = \bar{E}_i^r (1 - \phi_i^r)^2 \quad , \quad r = m, f, d \quad and \quad i = 1, 2, 3 \qquad (1.49)$$

where $\bar{E}_i^r$ are the effective moduli of elasticity along the principal axes defined along the direction of the fibers and transversely to them.

The damaging state is any state that satisfies $g = 0$. The four states are outlined below

$$g^r < 0, \qquad \text{(elastic unloading)} \qquad (1.50a)$$

$$g^r = 0, \qquad \frac{\partial g^r}{\partial \boldsymbol{Y}^r} : \dot{\boldsymbol{Y}}^r < 0 \quad \text{(elastic unloading)} \qquad (1.50b)$$

$$g^r = 0, \qquad \frac{\partial g^r}{\partial \boldsymbol{Y}^r} : \dot{\boldsymbol{Y}}^r = 0 \quad \text{(neutral loading)} \qquad (1.50c)$$

$$g^r = 0, \qquad \frac{\partial g^r}{\partial \boldsymbol{Y}^r} : \dot{\boldsymbol{Y}}^r > 0 \quad \text{(loading from a damaging state)} \qquad (1.50d)$$

11

1.6 Damage Evolution

1.6.1 Matrix Damage and Plasticity Evolution

The two energy dissipative mechanisms of plasticity and damage are exhibited by the metal matrix. The two energy dissipative behaviors influence each other. As out lined later in this section the plastic strain rate and the damage rate are each functions of the stress and the conjugate force to damage. Consequently, the energies dissipated due to damage and that due to plasticity are interdependent to each other.

The total power of dissipation of the matrix is given by

$$\tilde{\Pi}^m = \tilde{\Pi}^{md} + \tilde{\Pi}^{mp} \tag{1.51}$$

where $\tilde{\Pi}^{mp}$ is the plastic dissipation and $\tilde{\Pi}^{md}$ is the corresponding damage dissipation. The plastic dissipation is expressed as

$$\tilde{\Pi}^{mp} = \tilde{\boldsymbol{\sigma}}^m : \dot{\tilde{\boldsymbol{\epsilon}}}^{m''} + \tilde{\boldsymbol{\alpha}}^m : \dot{\tilde{\boldsymbol{\beta}}}^m \tag{1.52}$$

The second term of expression (1.52) is associated with kinematic hardening. The associated damage dissipation is given by

$$\tilde{\Pi}^{md} = \boldsymbol{Y}^m : \dot{\boldsymbol{\phi}}^m + K^m \dot{\kappa}^m + \gamma^m \dot{\zeta}^m \tag{1.53}$$

The term $K^m \dot{\kappa}^m$ is associated with isotropic damage hardening. The third term in expression (1.53) is associated with kinematic hardening. As mentioned previously, it relates to interaction of cracks.

Utilizing the calculus of functions of several variables, one introduces two Lagrange multipliers Λ_1^m and Λ_2^m in order to form the function Ω^m such that

$$\Omega^m = \tilde{\Pi}^m - \dot{\Lambda}_1^m \tilde{f}^m - \dot{\Lambda}_2^m g^m \tag{1.54}$$

In equation (1.54), $\tilde{f}^m(\tilde{\boldsymbol{\sigma}}^m, \tilde{\boldsymbol{\alpha}}^m)$ is the plastic yield function of the matrix and $\tilde{\boldsymbol{\alpha}}^m$ is the back-stress tensor. Since $\boldsymbol{Y}^m$ is a function of $\boldsymbol{\phi}^m$ and $\boldsymbol{\sigma}^m$ (as it will be shown in (1.84)), the yield function may be expressed as follows $\tilde{f}^m(\boldsymbol{Y}^m, \boldsymbol{\phi}^m, \tilde{\boldsymbol{\alpha}}^m)$. For non-associative plasticity the yield function $\tilde{f}^m$ should be replaced by the corresponding plastic potential function. The damage potential g^m is a function of $\boldsymbol{Y}^m$ and κ^m or $\tilde{\boldsymbol{\sigma}}^m$, $\boldsymbol{\phi}^m$ and κ^m such that

$$g^m \equiv g^m(\tilde{\boldsymbol{\sigma}}^m, \boldsymbol{\phi}^m, \kappa^m) = 0 \quad or \quad g^m \equiv g^m(\boldsymbol{Y}^m, \kappa^m) = 0 \tag{1.55}$$

In order to extremize the function Ω^m, one uses the necessary conditions

$$\frac{\partial \Omega^m}{\partial \tilde{\boldsymbol{\sigma}}^m} = 0 \tag{1.56}$$

and

$$\frac{\partial \Omega^m}{\partial \boldsymbol{Y}^m} = 0 \tag{1.57}$$

12

Equations (1.56) and (1.57) yield the corresponding plastic stain rate and damage rate evolution equations respectively, which are coupled as shown below

$$\dot{\tilde{\epsilon}}^{m''} = \dot{\Lambda}_1^m \frac{\partial \tilde{f}^m}{\partial \tilde{\sigma}^m} + \dot{\Lambda}_2^m \frac{\partial g^m}{\partial \tilde{\sigma}^m} \tag{1.58}$$

and

$$\dot{\phi}^m = \dot{\Lambda}_1^m \frac{\partial \tilde{f}^m}{\partial Y^m} + \dot{\Lambda}_2^m \frac{\partial g^m}{\partial Y^m} \tag{1.59}$$

Equation (1.59) gives the increment of the damage from the damage potential g^m and the yield function f^m.

It is clear that coupling exists between the plastic strain rate and the damage rate in the matrix.

Using the consistency condition for the matrix yield function $\tilde{f}^m$ and the matrix damage g^m such that

$$\dot{\tilde{f}}^m = 0 \tag{1.60a}$$

or

$$\frac{\partial \tilde{f}^m}{\partial Y^m} : \dot{Y}^m + \frac{\partial \tilde{f}^m}{\partial \phi^m} : \dot{\phi}^m + \frac{\partial \tilde{f}^m}{\partial \tilde{\alpha}^m} : \dot{\tilde{\alpha}}^m = 0 \tag{1.60b}$$

and

$$\dot{g}^m = 0 \tag{1.61a}$$

or

$$\frac{\partial g^m}{\partial \tilde{\sigma}^m} : \dot{\tilde{\sigma}}^m + \frac{\partial g^m}{\partial \phi^m} : \dot{\phi}^m + \frac{\partial g^m}{\partial \kappa^m} : \dot{\kappa}^m = 0 \tag{1.61b}$$

Expressions (1.61) state that after an increment of damage, the volume element again must be in a damaging state. Making use of expression (1.48) in its incremental form

$$\dot{\kappa}^m = Y^m : \dot{\phi}^m \tag{1.62}$$

one can then express equation (1.61b) as follows

$$\frac{\partial g^m}{\partial \tilde{\sigma}^m} : \dot{\tilde{\sigma}}^m + \frac{\partial g^m}{\partial \phi^m} : \dot{\phi}^m + \frac{\partial g^m}{\partial \kappa^m} : Y^m : \dot{\phi}^m = 0 \tag{1.63}$$

A Prager-Zigler kinematic hardening evolution law is used in the damaged configuration $\tilde{C}^m$ such that

$$\dot{\tilde{\alpha}}^m = \dot{\tilde{\mu}}^m (\tilde{\sigma}^m - \tilde{\alpha}^m) \tag{1.64}$$

13

where $\dot{\tilde{\mu}}^m$ is defined by Voyiadjis and Park (1995)[7] such that

$$\dot{\tilde{\mu}}^m = 3 \ b \ \dot{\Lambda}_1^m \tag{1.65}$$

and b is the kinematic hardening parameter for the matrix. The corresponding yield fuction is given by

$$\tilde{f}^m \equiv \frac{3}{2}(\tilde{\boldsymbol{\sigma}}^m - \tilde{\boldsymbol{\alpha}}^m) : (\tilde{\boldsymbol{\sigma}}^m - \tilde{\boldsymbol{\alpha}}^m) - \tilde{\sigma}_0^{m^2} = 0 \tag{1.66}$$

from this equations one can conclude

$$\frac{\partial \tilde{f}^m}{\partial \tilde{\boldsymbol{\sigma}}^m} = 3(\tilde{\boldsymbol{\sigma}}^m - \tilde{\boldsymbol{\alpha}}^m) \tag{1.67}$$

and therefore

$$\dot{\boldsymbol{\alpha}}^m = \dot{\Lambda}_1^m \frac{\partial \tilde{f}^m}{\partial \tilde{\boldsymbol{\sigma}}^m} b \tag{1.68}$$

Substituting for $\dot{\boldsymbol{\phi}}^m$ from equation (1.59) into equations (1.60b) and (1.63) and substituting $\dot{\tilde{\boldsymbol{\alpha}}}^m$ from equation (1.68) into equation (1.60b) one obtains the following linear system

$$\begin{pmatrix} \frac{\partial g^m}{\partial \tilde{\boldsymbol{\sigma}}^m} : \dot{\tilde{\boldsymbol{\sigma}}}^m \\ \frac{\partial f^m}{\partial \boldsymbol{Y}^m} : \dot{\boldsymbol{Y}}^m \end{pmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{pmatrix} \dot{\Lambda}_1^m \\ \dot{\Lambda}_2^m \end{pmatrix} \tag{1.69}$$

where the components a_{ij} are scalars given by

$$a_{11} = \left(\frac{\partial g^m}{\partial \boldsymbol{\phi}^m} + \frac{\partial g^m}{\partial \kappa^m} \boldsymbol{Y}^m \right) : \frac{\partial \tilde{f}^m}{\partial \boldsymbol{Y}^m} \tag{1.70a}$$

$$a_{12} = \left(\frac{\partial g^m}{\partial \boldsymbol{\phi}^m} + \frac{\partial g^m}{\partial \kappa^m} \boldsymbol{Y}^m \right) : \frac{\partial g^m}{\partial \boldsymbol{Y}^m} \tag{1.70b}$$

$$a_{21} = \left(\frac{\partial \tilde{f}^m}{\partial \boldsymbol{\phi}^m} : \frac{\partial \tilde{f}^m}{\partial \boldsymbol{Y}^m} + \frac{\partial \tilde{f}^m}{\partial \tilde{\boldsymbol{\alpha}}^m} : \frac{\partial \tilde{f}^m}{\partial \tilde{\boldsymbol{\sigma}}^m} b \right) \tag{1.70c}$$

$$a_{22} = \left(\frac{\partial \tilde{f}^m}{\partial \boldsymbol{\phi}^m} : \frac{\partial g^m}{\partial \boldsymbol{Y}^m} \right) \tag{1.70d}$$

Resolving the linear system (1.69) one obtains

$$\begin{pmatrix} \dot{\Lambda}_1^m \\ \dot{\Lambda}_2^m \end{pmatrix} = \frac{1}{\Delta} \begin{bmatrix} a_{22} & -a_{12} \\ -a_{21} & a_{11} \end{bmatrix} \begin{pmatrix} \frac{\partial g^m}{\partial \tilde{\boldsymbol{\sigma}}^m} : \dot{\tilde{\boldsymbol{\sigma}}}^m \\ \frac{\partial f^m}{\partial \boldsymbol{Y}^m} : \dot{\boldsymbol{Y}}^m \end{pmatrix} \tag{1.71}$$

where

$$\Delta = a_{11} a_{22} - a_{12} a_{21} \tag{1.72}$$

Substituting $\dot{\Lambda}_1^m$ and $\dot{\Lambda}_2^m$ from equation (1.71) into equation (1.59) one obtains

14

$$\dot{\phi}^m = \boldsymbol{S}^m : \dot{\tilde{\boldsymbol{\sigma}}}^m + \boldsymbol{R}^m : \dot{\boldsymbol{Y}}^m \tag{1.73}$$

where

$$\boldsymbol{S}^m = \frac{1}{\Delta} \frac{\partial g^m}{\partial \tilde{\boldsymbol{\sigma}}^m} : \left(a_{22} \frac{\partial \tilde{f}^m}{\partial \boldsymbol{Y}^m} + a_{21} \frac{\partial g^m}{\partial \boldsymbol{Y}^m} \right) \tag{1.74}$$

and

$$\boldsymbol{R}^m = \frac{1}{\Delta} \frac{\partial \tilde{f}^m}{\partial \boldsymbol{Y}^m} : \left(-a_{12} \frac{\partial \tilde{f}^m}{\partial \boldsymbol{Y}^m} + a_{11} \frac{\partial g^m}{\partial \boldsymbol{Y}^m} \right) \tag{1.75}$$

Since Y^m is a function of $\tilde{\boldsymbol{\sigma}}^m$ and $\boldsymbol{\phi}^m$ one can write

$$\dot{\boldsymbol{Y}}^m = \frac{\partial Y^m}{\partial \tilde{\boldsymbol{\sigma}}^m} : \dot{\tilde{\boldsymbol{\sigma}}}^m + \frac{\partial Y^m}{\partial \phi^m} : \dot{\phi}^m \tag{1.76}$$

substituting $\dot{\boldsymbol{Y}}^m$ from equation (1.76) into equation (1.73) one obtains the damage evolution equation in the matrix such that

$$\dot{\phi}^m = \boldsymbol{X}^m : \dot{\tilde{\boldsymbol{\sigma}}}^m \tag{1.77}$$

where $\boldsymbol{X}^m$ is a fourth order tensor originally defined in section 2 by equation (1.16). The explicit form of $\boldsymbol{X}^m$ is given below

$$\boldsymbol{X}^m = \left(\boldsymbol{S}^m + \boldsymbol{R}^m \frac{\partial Y^m}{\partial \tilde{\boldsymbol{\sigma}}^m} \right) \left(\boldsymbol{I}_4 - \boldsymbol{R}^m \frac{\partial Y^m}{\partial \phi^m} \right)^{-1} \tag{1.78}$$

where $\boldsymbol{I}_4$ is the fourth order identity tensor expressed as follows

$$I_{ijkl} = \frac{1}{2} (\delta_{ij}\delta_{kl} + \delta_{il}\delta_{jk}) \tag{1.79}$$

The thermodynamic force associated with the matrix damage is obtained by using the enthalpy of the damaged matrix as follows

$$V^m(\tilde{\boldsymbol{\sigma}}^m, \phi^m) = \frac{1}{2} \tilde{\boldsymbol{\sigma}}^m : \tilde{\boldsymbol{E}}^{-m}(\phi^m) : \tilde{\boldsymbol{\sigma}}^m - \Phi(\tilde{\boldsymbol{\alpha}}^m) \tag{1.80}$$

where $\Phi(\tilde{\boldsymbol{\alpha}}^m)$ is the specific energy due to kinematic hardening, and $\tilde{\boldsymbol{E}}^m$ is the elastic stiffness of the damaged matrix. Using equation (1.80) the thermodynamic force of the matrix is given by

$$\boldsymbol{Y}^m = \frac{\partial V^m}{\partial \phi^m} \tag{1.81}$$

Making use of the energy equivalence principle (Cordebois and Sidoroff, 1982)[11]

$$\bar{V}^m_{elastic} = \tilde{V}^m_{elastic} \tag{1.82}$$

15

one obtains a relation between the damaged elastic compliance, $\tilde{E}^{-m}$ for the matrix and the corresponding undamaged elastic compliance $\bar{E}^{-m}$ such that (Voyiadjis and Kattan, 1992a)[2]

$$\tilde{E}^{-m}(\phi^m) = M^m(\phi^m) : \bar{E}^{-m} : M^m(\phi^m) \tag{1.83}$$

Through the use of equations (1.80) and (1.81), the thermodynamic force for the matrix is obtained explicitly such that

$$Y_{ij}^m = (\tilde{\sigma}_{cd}^m \bar{E}_{abpq}^{-m} M_{pqkl}^m \tilde{\sigma}_{kl}^m) \frac{\partial M_{abcd}^m}{\partial \phi_{ij}^m} \tag{1.84}$$

1.6.2 Fiber Damage Evolution

For fibers the gradual degradation of the elastic stiffness is caused only through damage and no plastic dissipation occurs. The associated damage dissipation of the fiber is given by

$$\tilde{\Pi}^f = \tilde{\Pi}^{fd} = Y^f : \dot{\phi}^f + K^f \dot{\kappa}^f \tag{1.85}$$

and

$$\bar{\Pi}^f = 0 \tag{1.86}$$

Since the plastic dissipation is zero. The function Ω^f for the fiber is given by

$$\Omega^f = \tilde{\Pi}^f - \dot{\Lambda}^f g^f \tag{1.87}$$

and the corresponding damage rate evolution of the fiber is given by the expression

$$\dot{\phi}^f = \dot{\Lambda}^f \frac{\partial g^f}{\partial Y^f} \tag{1.88}$$

Making use of the consistency condition for the damage of the fiber

$$\dot{g}^f = 0 \tag{1.89}$$

one obtains the evolution equation for ϕ^f such that

$$\dot{\phi}^f = X^f : \dot{\tilde{\sigma}}^f \tag{1.90}$$

where X^f is a fourth order tensor similar to X^m expressed by equation (1.78). The thermodynamic force associated with damage of the fiber Y^f is obtained in similar approach to that of the matrix, Y^m and has a similar form except replacing the subscript m with f.

1.6.3 Interfacial Damage Evolution

The second order symmetric tensor ϕ^d is used to describe the interfacial damage as depicted in Figure 1.3 and expressed as folloes (Voyiadjis and Park, 1995[7])

$$\phi^d = \phi^d(S, \bar{S}) \tag{1.91}$$

the power of dissipation due to interfacial damage is given by

$$\Pi^d = Y^d : \dot{\phi}^f + K^f \dot{\kappa}^f \tag{1.92}$$

and

$$\tilde{\Pi}^d = 0 \tag{1.93}$$

The function Ω^d is expressed by

$$\Omega^d = \Pi^d - \dot{\Lambda}^d g^d \tag{1.94}$$

where

$$\dot{\phi}^d = \dot{\Lambda}^d \frac{\partial g^d}{\partial Y^d} \tag{1.95}$$

From the consistency condition for the interfacial damage one obtains the evolution equation for ϕ^d such that

$$\dot{\phi}^d = X^d : \dot{\tilde{\sigma}}^d \tag{1.96}$$

The corresponding thermodynamic force for interfacial damage is obtained using a similar procedures to that outlined for the two types of damage and is given by

$$Y_{ij}^d = (\sigma_{cd} \tilde{E}_{abpq}^{-1} M_{pqkl}^d \sigma_{kl}) \frac{\partial M_{abcd}^d}{\partial \phi_{ij}^d} \tag{1.97}$$

1.7 Stiffness Tensor for the Model

In order to obtain the elasto-plastic stiffness tensor for the damaged composite system the following procedure is followed. Separate constitutive equations for each of the matrix and the fiber are first derived in their respective damaged configurations $\tilde{C}^m$ and $\tilde{C}^f$. These two constitutive equations are then combined into one in order to express the overall composite system in its partial effective configuration $\tilde{C}$. The interfacial damage is finally incorporated into the system in order to obtain the final constitutive equation and its corresponding stiffness tensor that includes all three types of damages in the damaged configuration C.

The elasto-plastic behavior of the matrix and the elastic behavior of the fiber in their respective effective configurations $\bar{C}^m$ and $\bar{C}^f$ are given as follows

$$\dot{\bar{\sigma}}^m = \bar{D}^m : \dot{\bar{\epsilon}}^m \tag{1.98}$$

17

and

$$\dot{\bar{\sigma}}^f \;=\; \bar{E}^f : \dot{\bar{\epsilon}}^f \tag{1.99}$$

where $\bar{D}^m$ and $\bar{E}^f$ are, respectively, the fourth order elasto-plastic stiffness tensor of the matrix and the elastic stiffness tensor of the fiber material. The stiffness $\bar{D}^m$ is given by Voyiadjis and Kattan(1992b) [12] such that

$$\bar{D}^m \;=\; \bar{E}^m - \frac{1}{\bar{Q}^m}\left(\frac{\partial \bar{f}^m}{\partial \bar{\sigma}^m} : \bar{E}^m\right) \otimes \left(\bar{E}^m : \frac{\partial \bar{f}^m}{\partial \bar{\sigma}^m}\right) \tag{1.100}$$

where

$$\bar{Q}^m \;=\; \frac{\partial \bar{f}^m}{\partial \bar{\sigma}^m} : \bar{E}^m : \frac{\partial \bar{f}^m}{\partial \bar{\sigma}^m} - b\frac{\partial \bar{f}^m}{\partial \bar{\alpha}^m} : (\bar{\sigma}^m - \bar{\alpha}^m)\frac{\frac{\partial \bar{f}^m}{\partial \bar{\sigma}^m} : \frac{\partial \bar{f}^m}{\partial \bar{\sigma}^m}}{(\bar{\sigma}^m - \bar{\alpha}^m) : \frac{\partial \bar{f}^m}{\partial \bar{\sigma}^m}} \tag{1.101}$$

and b is a material constant relating to the kinematic hardening of the matrix. The corresponding yield function is given by

$$\bar{f}^m \equiv \frac{3}{2}(\bar{\sigma}^m - \bar{\alpha}^m) : (\bar{\sigma}^m - \bar{\alpha}^m) - \sigma_0^{m^2} = 0 \tag{1.102}$$

A kinematic hardening law of the Prager-Ziegler Type is used in conjunction with this work such that

$$\dot{\bar{\alpha}}^m \;=\; \dot{\bar{\mu}}^m(\bar{\sigma}^m - \bar{\alpha}^m) \tag{1.103}$$

where $\dot{\bar{\mu}}^m$ is a local scalar multiplier.

The component damaged elastic stiffness tensors $\tilde{E}^m$ and $\tilde{E}^f$ in the local configurations $\tilde{C}^m$ and $\tilde{C}^f$ respectively are given by (Voyiadjis and Kattan, 1993)[6]

$$\tilde{E}^m \;=\; M^{-m} : \bar{E}^m : M^{-m} \tag{1.104}$$

and

$$\tilde{E}^f \;=\; M^{-f} : \bar{E}^f : M^{-f} \tag{1.105}$$

The overall response of the composite system in the partial effective configuration, $\tilde{C}$, is given by

$$\dot{\tilde{\sigma}} \;=\; \tilde{D} : \dot{\tilde{\epsilon}} \tag{1.106}$$

In order to obtain $\tilde{D}$ use is made of the following relations in the partial effective configuration, $\tilde{C}$

$$\dot{\tilde{\sigma}} \;=\; \tilde{c}^m \dot{\tilde{\sigma}}^m + \tilde{c}^f \dot{\tilde{\sigma}}^f \tag{1.107}$$

$$\dot{\tilde{\sigma}}^m \;=\; \tilde{D}^m : \dot{\tilde{\epsilon}}^m \tag{1.108}$$

$$\dot{\tilde{\sigma}}^f \;=\; \tilde{E}^f : \dot{\tilde{\epsilon}}^f \tag{1.109}$$

$$\dot{\tilde{\epsilon}}^m \;=\; \tilde{A}^m : \dot{\tilde{\epsilon}} \tag{1.110}$$

$$\dot{\tilde{\epsilon}}^f \;=\; \tilde{A}^f : \dot{\tilde{\epsilon}} \tag{1.111}$$

The expression for $\tilde{\boldsymbol{D}}$ is given by

$$\tilde{\boldsymbol{D}} = \tilde{c}^m \tilde{\boldsymbol{D}}^m : \tilde{\boldsymbol{A}}^m + \tilde{c}^f \tilde{\boldsymbol{E}}^f : \tilde{\boldsymbol{A}}^f \tag{1.112}$$

$\tilde{\boldsymbol{D}}^m$ is the elasto-plastic stiffness for the damaged matrix constituent. Equation (1.98) is transformed from the undamaged matrix configuration $\bar{C}^m$ to the damaged matrix configuration $\tilde{C}^m$ in order to obtain the damaged elasto-plastic stiffness of the matrix constituent, $\tilde{\boldsymbol{D}}^m$. Use is made of equation (1.27) together with its strain rate counterpart and substituted in equation (1.98) in order to obtain $\tilde{\boldsymbol{D}}^m$, such that

$$\tilde{\boldsymbol{D}}^m = \boldsymbol{m}^{-m} : \bar{\boldsymbol{D}}^m : \boldsymbol{m}^{-m} \tag{1.113}$$

In order to obtain the overall damage response of the composite system the interfacial damage tensor $\boldsymbol{m}^d$ needs to be incorporated in order to transform $\tilde{\boldsymbol{D}}$ from the configuration $\tilde{C}$ to the overall damaged configuration C.

The elasto-plastic stiffness of the damaged material is given by

$$\dot{\boldsymbol{\sigma}} = \boldsymbol{D} : \dot{\boldsymbol{\epsilon}} \tag{1.114}$$

where

$$\boldsymbol{D} = \boldsymbol{m}^{-d} : \tilde{\boldsymbol{D}} : \boldsymbol{m}^{-d} \tag{1.115}$$

The corresponding elastic stiffness $\boldsymbol{E}$ for the damaged composite is such that

$$\boldsymbol{E} = \boldsymbol{M}^{-d} : \tilde{\boldsymbol{E}} : \boldsymbol{M}^{-d} \tag{1.116}$$

where the elastic stiffness in the partial effective configuration $\tilde{C}$ is given by

$$\tilde{\boldsymbol{E}} = \tilde{c}^m \tilde{\boldsymbol{E}}^m : \tilde{\boldsymbol{A}}^{me} + \tilde{c}^f \tilde{\boldsymbol{E}}^f : \tilde{\boldsymbol{A}}^{fe} \tag{1.117}$$

1.8 Evolution of Material Parameters for the Case of Isotropic Damage

The special case of isotropic damage is investigated here.

For the special case of uniaxial monotonic loading for isotropic materials the complementary strain energy is expressed as follows

$$V = \frac{\sigma^2}{2E} \tag{1.118}$$

or

$$V = \frac{1}{2\bar{E}(1-\phi)^2}\sigma^2 \tag{1.119}$$

where $\bar{E}$ is the elastic stiffness of the isotropic material and $\phi = \phi_{11} = \phi_{22} = \phi_{33}$. Making use of equation (1.118) and

$$Y = \frac{\partial V}{\partial \phi} \tag{1.120}$$

one obtains the conjugate stress for damage such that

$$Y = \frac{1}{\bar{E}(1-\phi)^3}\sigma^2 \tag{1.121}$$

The damage criterion used here is simplified such that c is set equal to zero and therefore

$$Y_{ij}P_{ijkl}Y_{kl} - 1 = 0 \tag{1.122}$$

where

$$P_{ijkl} = (u_{ij} + V_{ij})(u_{kl} + V_{kl}) \tag{1.123}$$

which is not a direct function of damage ϕ as in equation (1.45).
The corresponding damage criterion is given by

$$g \equiv \frac{\bar{E}^{-2}(1-\phi)^{-6}\sigma^4}{\left[\lambda\eta\left(\frac{\kappa}{\lambda}\right)^\xi + V_{11}\right]^2} - 1 = 0 \tag{1.124}$$

and

$$V_{11} = \lambda v^2 \tag{1.125}$$

At initiation of damage, $g = 0$ and $\phi = 0$, consequently one obtains

$$\sigma_0 = v\sqrt{\lambda\bar{E}} \tag{1.126}$$

where σ_0 is the uniaxial stress at initiation of damage.

In Figure 1.4 the damage criterion g versus σ is plotted. Three values of σ_0 are outlined for different cases of v while $\eta = 0.08$ and $\xi = 0.55$. The increase in the magnitude of v delays the onset of damage. Small stress increments of 0.01 MPa were applied in order to ensure convergence of the solution.

The power of dissipation of damage in this simple case is given as

$$\Pi = Y\dot{\phi} + K\dot{\kappa} + \sigma\dot{\epsilon}" + \alpha\dot{\beta} \tag{1.127}$$

Using the consistency condition one can solve for $\dot{\phi}$ and obtain

$$d\phi = \frac{-\frac{\partial g}{\partial \sigma}d\sigma}{\frac{\partial g}{\partial \phi} + \frac{\partial g}{\partial \kappa}Y} \tag{1.128}$$

Figure 1.5 shows the behavior of damage for different values of the damage parameter v. It is clear that smaller values of v introduce damage and failure at lower values of stress, σ.

Figures 1.6 and 1.7 show respectively the variation of damage with η and ξ for a constant value of v.

A consistent incremental damage theory is presented in this work that allows damage to be path dependent with respect to the damage conjugate stress Y.

Table 1.4: Local Damage Parameters

	Matrix Damage	Fiber Damage	Interfacial Damage
η_1	0.08	0.06	0.075
η_2	0.08	0.06	0.073
η_3	0.08	0.06	0.073
ξ_1	0.55	0.52	0.55
ξ_2	0.55	0.52	0.55
ξ_3	0.55	0.52	0.55
v_1	0.0013	0.001	0.004
v_2	0.0013	0.001	0.003
v_3	0.0013	0.001	0.003

1.9 Evolution of Different Types of Damage for a Composite Material Loaded in the Fiber Direction

In this case, a metal matrix composite is loaded monotonically in the elastic region, along the fiber direction. Following the proposed constitutive model for damage presented in this work, the evolution of the different types of damage versus stress is presented in Figure 1.8. The types of damage considered here are the matrix damage, ϕ^m, the fiber damage, ϕ^f, and the debonding damage, ϕ^d, Their respective evolutions and behaviors are dictated by the experimentally determined material parameters shown in Table 1.4 (Voyiadjis and Venson,1995 [10]; Voyadjis and Park, 1995 [7]). As depicted in Figure 1.8, at a stress level of $800MPa$, the respective damages are $\phi^f = 0.005, \phi^m = 0.013$ and $\phi^d = 0.061$. The order of initiation and evolution of damage for each constituent is dictated by the material parameters obtained experimentally.

Figure 1.9 shows the degradation of the matrix and the fiber stiffness due to the loading described above, while Figure 1.10 presents the stress-strain curves for matrix, the fiber and the whole composite utilizing the Mori-Tanaka homogenization procedure.

21

1.10 Conclusion

A consistent coupled incremental damage and plasticity theory is presented in this work that allows damage to be path dependent either on the stress history or the thermodynamic force conjugate to damage. This coupling occurs only in the matrix since the fibers are assumed to be only elastic. This is achieved through the use of incremental damage tensors. Damage and plastic deformations are incorporated in the proposed model that is used for the analysis of fiber-reinforced metal matrix composite materials. The proposed micro-mechanical damage relations are used for each of the matrix and the fiber. This is coupled with the interfacial damage between the matrix and the fiber exclusively. The damage relations are linked to the overall response through a homogenization procedure.

Evolution of damage is performed for both the cases of overall isotropic damage and damage of the individual constituents. The material parameters used here in order to quatify damage, are quite versatile and adequate in describing the physical evolution of damage in the composite constituents. This is clearly indicated in Figures 1.5 - 1.8.

In order to resolve more complicated problems than those presented here, the finite element method is required for determining the evolution of damage. The finite element implementation is presentedin Chapter 3 and numerical results for cyclic damage are presented in Chapter 4.
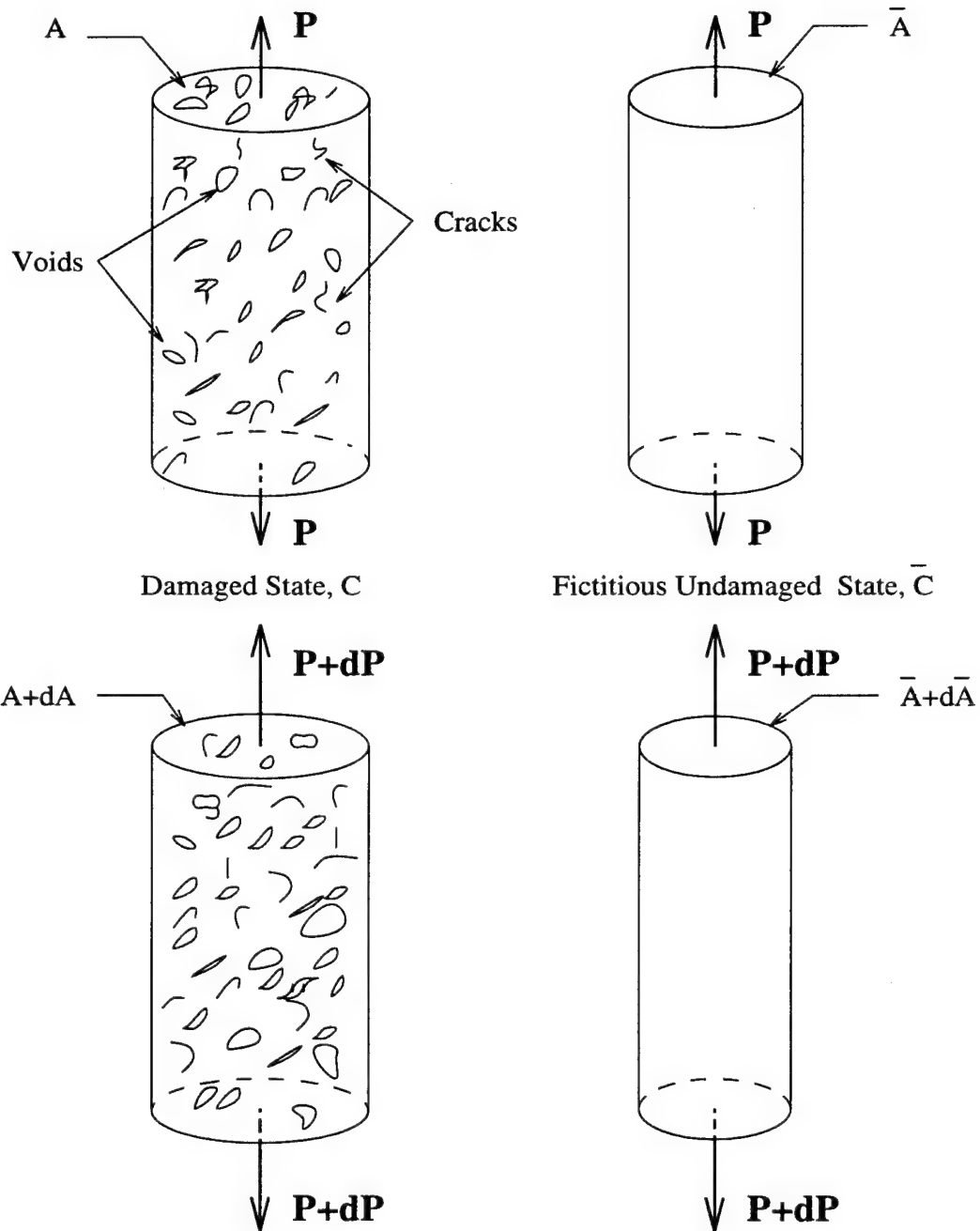
Figure 1.1: Schematic representation of damage due to uniaxial loading

The labels within the figure read:

A

P

Voids

Cracks

P

Damaged State, C

P

$\bar{A}$

P

Fictitious Undamaged State, $\bar{C}$

P+dP

A+dA

P+dP

P+dP

$\bar{A}+d\bar{A}$

P+dP
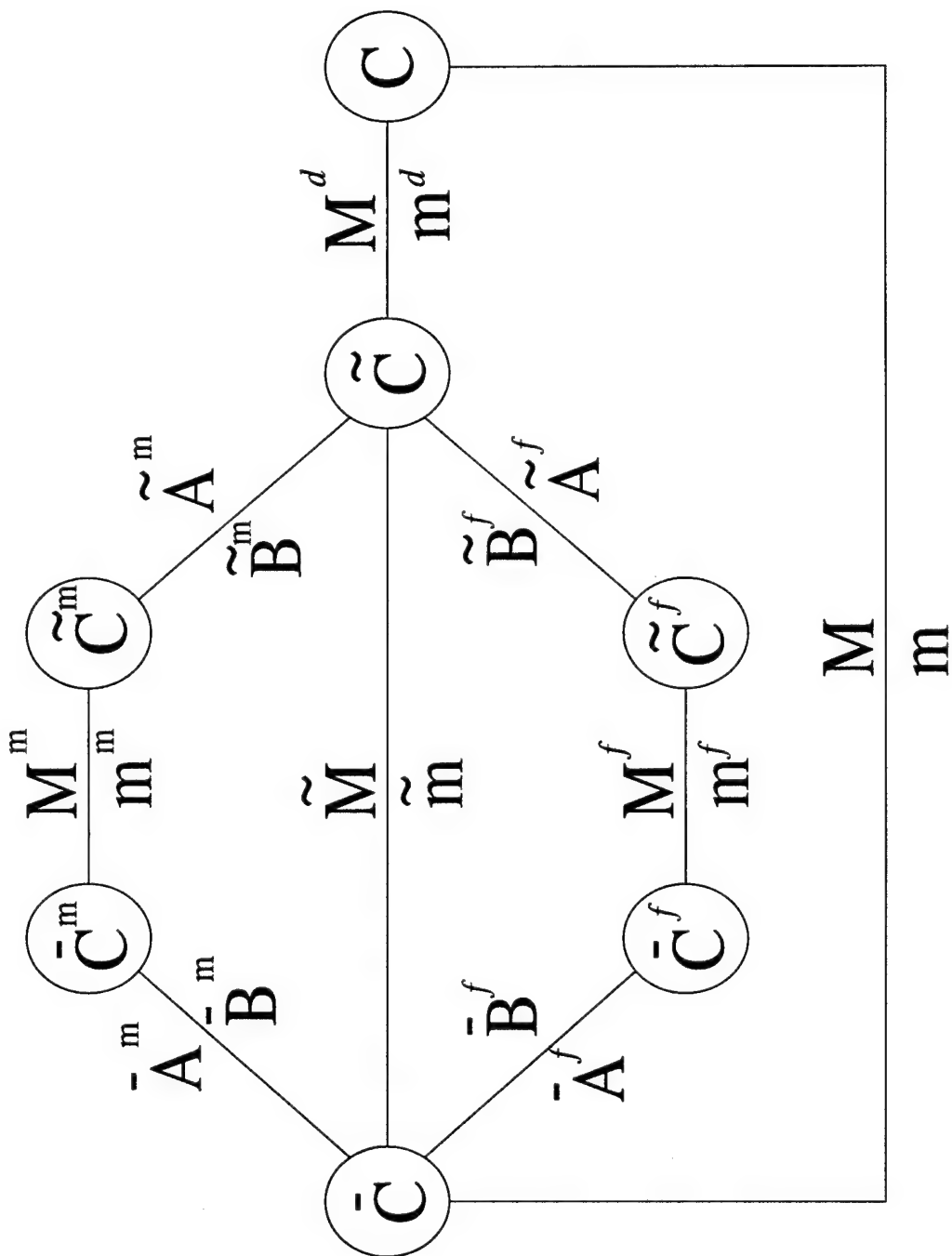
Figure 1.2: Different configurations of the composite

24
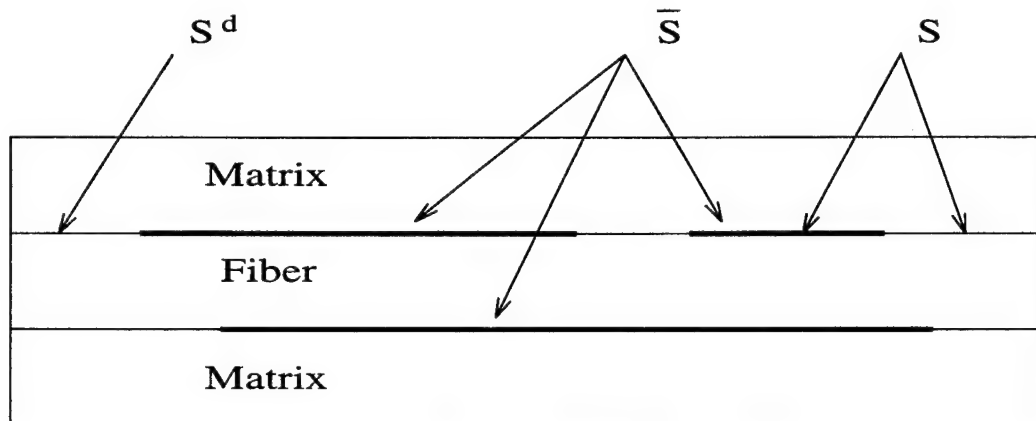
$$\phi_{22}^{d} = \frac{S - \overline{S}}{S}$$
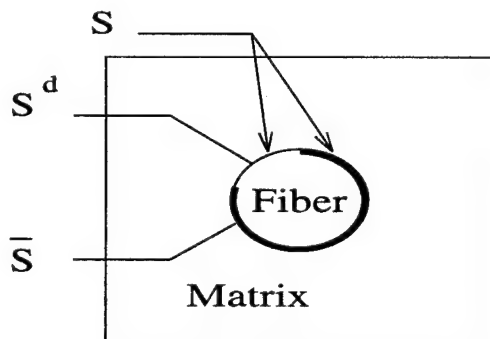
$$\phi_{11}^{d} = \frac{S - \overline{S}}{S}$$

$\overline{S}$ = Net resisting lenght

S = Total interfacial lenght

S^{d} = Debonding lenght

Figure 1.3: Schematic representation of interfacial damage
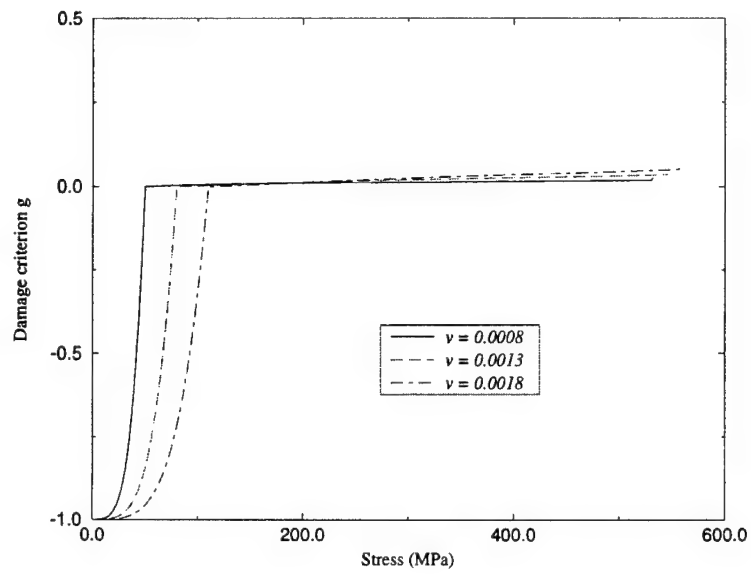
Figure 1.4: Damage criterion g for different damage parameters ν
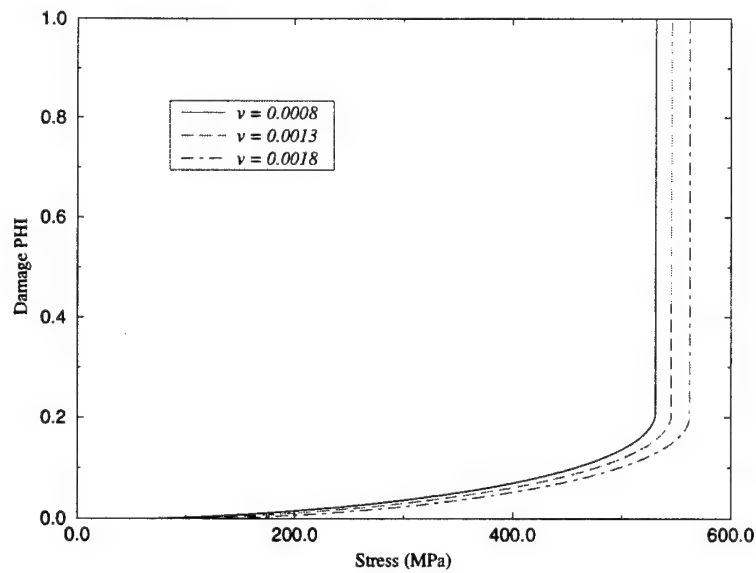
Figure 1.5: Damage evolution for different damage parameters ν

26
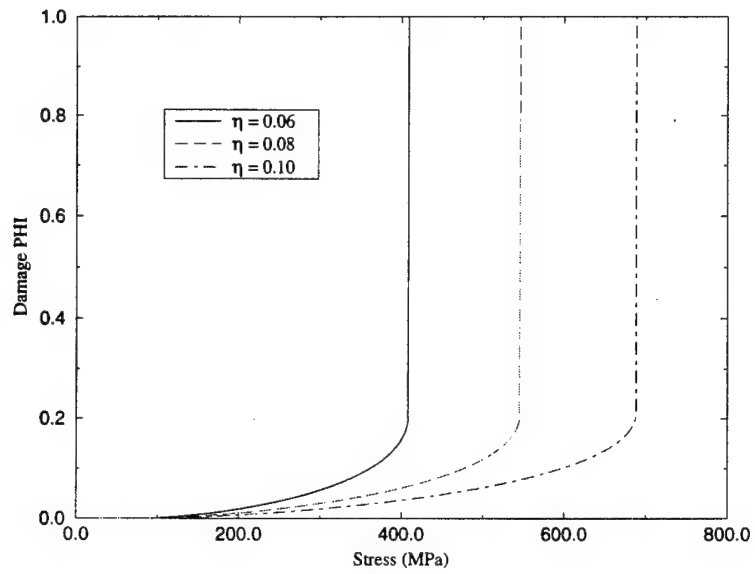
Figure 1.6: Damage evolution for different damage parameters η
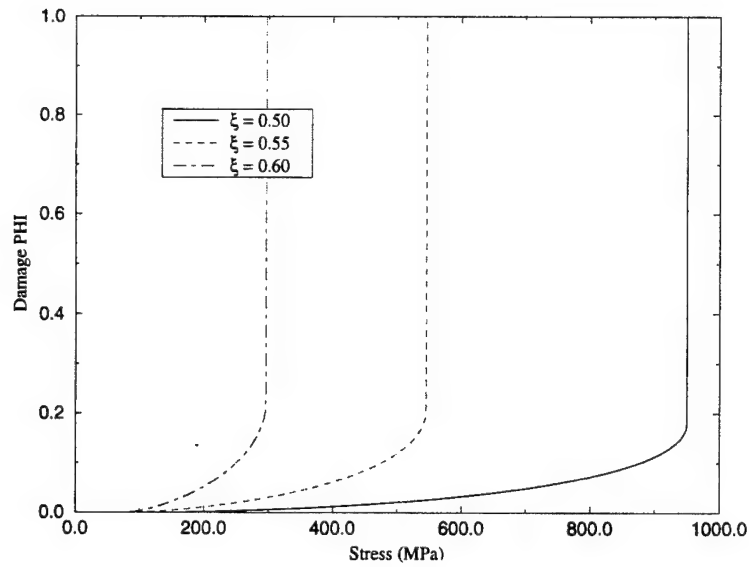
Figure 1.7: Damage evolution for different damage parameters ξ
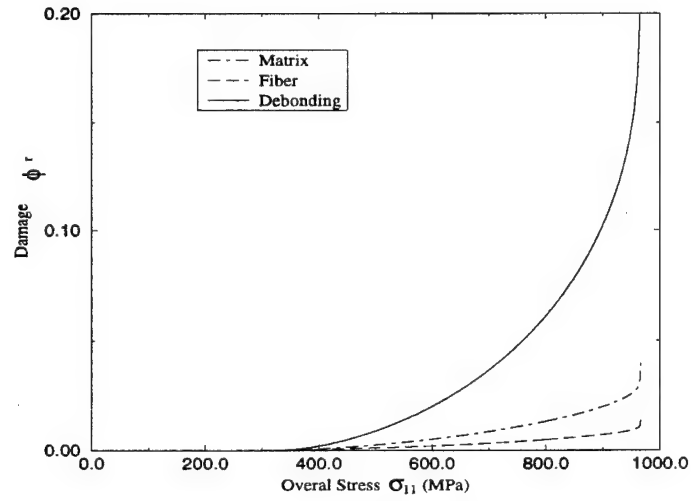
Figure 1.8: Matrix, fiber and interfacial damage evolutions
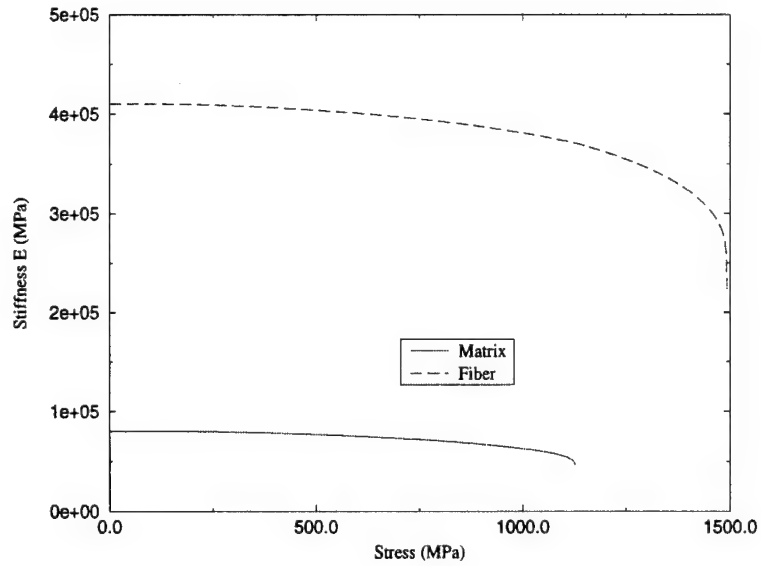
Figure 1.9: Matrix and fiber Stiffness evolutions

28
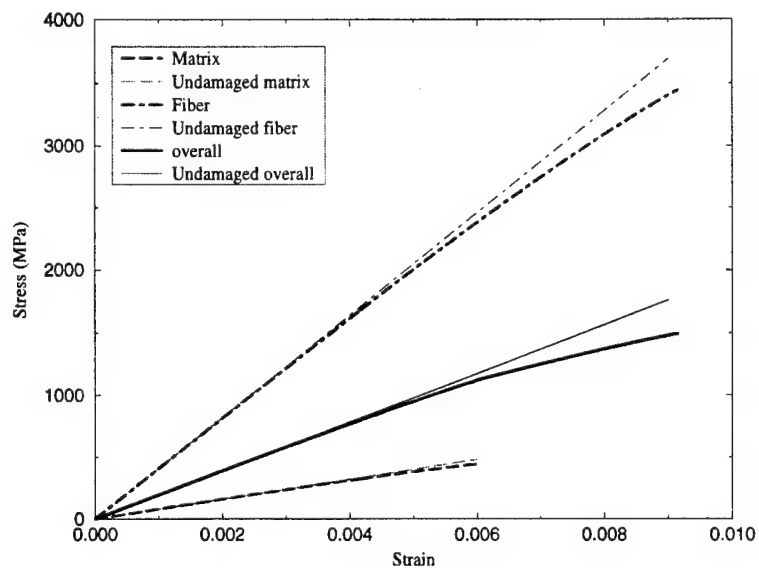
Figure 1.10: Matrix, fiber and overall Stress-Strain curves

29

Chapter 2

Elasto-Plastic Stress and Strain Concentration Tensors For Damaged Fibrous Composites

2.1 Introduction

As composite materials undergo damage, the corresponding stress and strain concentration tensors do not remain constant even for the case of elastic deformations. These tensors are derived and presented here for fibrous composites that undergo damage in both the constituents and the interfacial surface. The damage in the matrix includes nucleation and growth of voids, etc., while in the fibers includes micro-fracture, etc. In addition, interfacial damage is described as the damage due to debonding. The damaged stress and strain concentration tensors are obtained for the elasto-plastic states of the metal matrix composite material. Expressions are derived and presented here for the elasto-plastic stress and strain concentration tensors for fibrous metal matrix composite materials in the damaged configurations. The fibers are assumed to be continuous in this work. The elastic strain and stress concentration tensors using the Mori-Tanaka method for the case of undamaged fiberous composites are formulated by Chen, et al. (1992)[13]. These elastic concentration tensors are constant in the undamaged or effective configuration due to the fact that damage effects in the material are ignored. However, these damaged elasto-plastic concentration tensors are a function of the fourth order damage tensors in the damaged configuration.

2.2 Theoretical Preliminaries

In the initial configuration, C_0 , the composite material is assumed to be undeformed and undamaged. The initial matrix and fiber subconfigurations are denoted by C_0^m and C_0^f, respectively. Due to applied loads, the composite material is assumed to undergo damage and the resulting overall configuration is denoted by C. The resulting matrix and fiber local subconfigurations are denoted by C^m and C^f,, respectively. Damage is quantified using the concept proposed by Kachanov (1958)[1] whereby two kinds of fictitious configurations

30

$\bar{C}$ and $\tilde{C}$ of the composite systems are considered. $\bar{C}$ configuration is obtained from by removing all damages, while $\tilde{C}$ configuration is obtained from by removing only interfacial damage between the matrix and the fiber. $\bar{C}$ is termed the full effective configuration, while $\tilde{C}$ is the partial effective configuration. The matrix and fiber subconfigurations of the full effective configuration are denoted respectively by $\bar{C}^m$ and $\bar{C}^f$. Similarly, $\tilde{C}^m$ and $\tilde{C}^f$ are the subconfigurations of the partial effective configuration. This is shown in Figure 2.1.

The basic feature of the approach presented here is that local effects of damages are considered whereby these effects are described separately by the matrix, fiber, and interfacial damage tensors. The fourth order matrix damage effect tensor $\boldsymbol{M}^m$ and the fourth order fiber damage effect tensor $\boldsymbol{M}^f$ are defined such that

$$\bar{\sigma}_{ij}^m = M_{ijkl}^m \tilde{\sigma}_{kl}^m \tag{2.1}$$

and

$$\bar{\sigma}_{ij}^f = M_{ijkl}^f \tilde{\sigma}_{kl}^f \tag{2.2}$$

where $\bar{\boldsymbol{\sigma}}^m$ and $\bar{\boldsymbol{\sigma}}^f$ are the full effective matrix and fiber stresses in the subconfigurations $\bar{c}^m$ and $\bar{c}^f$, respectively, while $\tilde{\boldsymbol{\sigma}}^m$ and $\tilde{\boldsymbol{\sigma}}^f$ are the partial effective matrix and fiber stresses in the subconfigurations $\tilde{c}^m$, $\tilde{c}^f$ respectively. In addition, the fourth order interfacial damage effect tensor $\boldsymbol{M}^d$ is defined such as

$$\tilde{\sigma}_{ij} = M_{ijkl}^d \sigma_{kl} \tag{2.3}$$

where $\tilde{\boldsymbol{\sigma}}$ is the partial effective composite stress in the $\tilde{C}$ configuration, while $\boldsymbol{\sigma}$ is the overall stress of the composite in the C configuration. The respective damage effect tensors are clearly indicated in Figure 2.1. The local damage effect tensors $\boldsymbol{M}^m$ and $\boldsymbol{M}^f$ encompass all the pertinent damages related to the matrix and fiber, respectively, while the damage effect tensor $\boldsymbol{M}^d$ reflects the damage pertinent to the interfacial damage such as debonding (Voyiadjis and Park, 1995)[7].

2.3 Theoretical Formulation of the Overall Damage Effect Tensor M

Considering the overall configurations $C, \tilde{C}$ and $\bar{C}$, one can introduce an overall damage effect tensor $\boldsymbol{M}$ and a partial damage effect tensor $\tilde{\boldsymbol{M}}$ for the whole composite system. These tensors are defined similarly to the definitions of $\boldsymbol{M}^m$, $\boldsymbol{M}^f$ and $\boldsymbol{M}^d$ such that

$$\bar{\sigma}_{ij} = M_{ijkl} \sigma_{kl} \tag{2.4}$$

and

$$\bar{\sigma}_{ij} = \tilde{M}_{ijkl} \tilde{\sigma}_{kl} \tag{2.5}$$

The tensor $\boldsymbol{M}$ reflects all types of damages that the composite undergoes including the damage due to the interaction between the matrix and fibers, while the tensor $\tilde{\boldsymbol{M}}$ reflects

the damage in the matrix and fiber excluding the interfacial damage. This tensor has been studied previously by Voyiadjis and Kattan (1993)[6]. A matrix representation was explicitly derived for this fourth order tensor by expressing the stresses in vector form. The tensor M was shown to be symmetric. The symmetry property of the tensor M is used extensively in the derivation that follows. The same holds true for the tensors M^m, M^f and M^d. Similar to tensor M^d, both tensors M^m and M^f could be represented in terms of second order damage variable tensors ϕ^m and ϕ^f respectively, as shown by Voyiadjis and Park (1995)[7]. The overall effective composite stress $\bar{\sigma}$ is related to the local effective stresses $\bar{\sigma}^m$ and $\bar{\sigma}^f$ by making use of the micromechanical model proposed by Dvorak and Bahei-El-Din (1979)[14] such that

$$\bar{\sigma}_{ij} = \bar{c}^m \bar{\sigma}_{ij}^m + \bar{c}^f \bar{\sigma}_{ij}^f \tag{2.6}$$

where $\bar{c}^m$ and $\bar{c}^f$ are the matrix and fiber volume fractions, respectively in the undamaged configurations. The partial effective stresses of the matrix and fiber are related to the partial effective overall stress of the composite by the partial damage stress concentration tensors, such that

$$\tilde{\sigma}_{kl}^m = \tilde{B}_{klpq}^m \tilde{\sigma}_{pq} \tag{2.7}$$

$$\tilde{\sigma}_{kl}^f = \tilde{B}_{klpq} \tilde{\sigma}_{pq} \tag{2.8}$$

where $\tilde{B}^m$ and $\tilde{B}^f$ are the partial damaged stress concentration tensors for the matrix and fibers, respectively. Substituting equations (2.1) and (2.2) into (2.6), one obtains the following expression:

$$\bar{\sigma}_{ij} = \bar{c}^m M_{ijkl}^m \tilde{\sigma}_{kl}^m + \bar{c}^f M_{ijkl}^f \tilde{\sigma}_{kl}^f \tag{2.9}$$

Making use of (2.7) and (2.8) into (2.9), one obtains the following expression:

$$\bar{\sigma}_{ij} = \bar{c}^m M_{ijkl}^m \tilde{B}_{klpq}^m \tilde{\sigma}_{pq} \tag{2.10}$$

or

$$\bar{\sigma}_{ij} = (\bar{c}^m M_{ijkl}^m \tilde{B}_{klpq}^m + \bar{c}^f M_{ijkl}^f \tilde{B}_{klpq}^f)\tilde{\sigma}_{pq} \tag{2.11}$$

Finally substituting equation (2.3) into (2.11), one obtains the following relation:

$$\bar{\sigma}_{ij} = (\bar{c}^m M_{ijkl}^m \tilde{B}_{klpq}^m + \bar{c}^f M_{ijkl}^f \tilde{B}_{klpq}^f)M_{pqrs}^d \sigma_{rs} \tag{2.12}$$

Comparing equation (2.4) with equation (2.12), the following relation is obtained

$$M_{ijkl} = (\bar{c}^m M_{ijpq}^m \tilde{B}_{pqrs}^m + \bar{c}^f M_{ijpq}^f \tilde{B}_{pqrs}^f)M_{rskl}^d \tag{2.13}$$

or

$$M_{ijkl} = \tilde{M}_{ijrs}M_{rskl}^d \tag{2.14}$$

where

$$\tilde{M}_{ijrs} = \bar{c}^m M_{ijpq}^m \tilde{B}_{pqrs}^m + \bar{c}^f M_{ijpq}^f \tilde{B}_{pqrs}^f \tag{2.15}$$

2.4 Effective Volume Fraction

Since the fictitious effective configuration is obtained by removing all damages that the material has been subjected to consequently it follows that the volume fractions in the effective configuration will differ from the initial volume fractions. However, the volume fractions of configuration C are assumed to be equal to the initial volume fractions. In order to obtain an evolution expression for the effective volume fractions, we first address the simple case of one-dimensional damage model using the definition of Kachanov's (1958)[1] effective stress concept. The effective local stresses for the matrix and fiber in the one-dimensional case are defined by

$$\bar{\sigma}^m = \frac{1}{1 - \phi^m} \tilde{\sigma}^m \tag{2.16}$$

and

$$\bar{\sigma}^f = \frac{1}{1 - \phi^f} \tilde{\sigma}^f \tag{2.17}$$

where

$$\phi^m = \frac{d\tilde{A}^m - d\bar{A}^m}{d\tilde{A}^m} \tag{2.18}$$

and

$$\phi^f = \frac{d\tilde{A}^f - d\bar{A}^f}{d\tilde{A}^f} \tag{2.19}$$

where $d\tilde{A}^r$ and $d\bar{A}^r$ are differential areas normal to the fiber direction in the configuration $\tilde{C}$, and $\bar{C}$, respectively, where $r = m$ or f. The corresponding initial volume fractions are defined as follows:

$$c_0^r = \frac{dA_0^r}{dA_0} \tag{2.20}$$

where

$$dA_0 = dA_0^m + dA_0^f \tag{2.21}$$

Similarly, the effective volume fractions can be defined such as

$$\bar{c}^r = \frac{d\bar{A}^r}{d\bar{A}} \tag{2.22}$$

and

$$d\bar{A} = d\bar{A}^m + d\bar{A}^f \tag{2.23}$$

From relations (2.18) and (2.19), one obtains respectively,

$$d\bar{A}^m = (1 - \phi^m)d\tilde{A}^m \tag{2.24}$$

and

$$d\bar{A}^f = (1 - \phi^f)d\tilde{A}^f \tag{2.25}$$

Substituting relations (2.24) and (2.25) into equation (2.22) and making use of the assumption, $c_0^r = c^r$, one obtains the following relations

$$\frac{dA^f}{dA^m} = \frac{dA_0^f}{dA_0^m} \tag{2.26}$$

one obtains the following relations

$$\bar{c}^m = \frac{1 - \phi^m}{(1 - \phi^m) + (1 - \phi^f)\frac{c_0^f}{c_0^m}} \tag{2.27}$$

and

$$\bar{c}^f = \frac{1 - \phi^f}{(1 - \phi^f) + (1 - \phi^m)\frac{c_0^m}{c_0^f}} \tag{2.28}$$

Equations (2.27) and (2.28) satisfy the constraint

$$\bar{c}^m + \bar{c}^f = 1 \tag{2.29}$$

The variation of the effective volume fractions with matrix and fiber damage are shown in Figures 2.2 and 2.3, respectively, for the uniaxially loaded lamina. The initial fiber volume fraction is set equal to 0.35.

The generalization of equations (2.27) and (2.28) to the three-dimensional damage model using the second order damage tensor ϕ may be expressed as follows:

$$\bar{c}^m = \frac{1 - \phi_{eq}^m}{(1 - \phi_{eq}^m) + (1 - \phi_{eq}^f)\frac{c_0^f}{c_0^m}} \tag{2.30}$$

and

$$\bar{c}^f = \frac{1 - \phi_{eq}^f}{(1 - \phi_{eq}^f) + (1 - \phi_{eq}^m)\frac{c_0^m}{c_0^f}} \tag{2.31}$$

where

$$\phi_{eq}^m = \frac{(\phi_{ij}^m \phi_{ij}^m)^{1/2}}{\phi_{cr}^m} \tag{2.32}$$

and

$$\phi_{eq}^f = \frac{(\phi_{ij}^f \phi_{ij}^f)^{1/2}}{\phi_{cr}^f} \tag{2.33}$$

where ϕ_{cr}^m and ϕ_{cr}^f are the critical values of ϕ_{eq}^m and ϕ_{eq}^f, respectively, at failure.

2.5 Partial Damaged Stress and Strain Concentration Tensors

The matrix and fiber stress concentration factors are defined as fourth-rank tensors. As composites undergo damage, the stress and strain concentration factors do not remain constant. The relations for the effective-elastic stress concentration factors for the matrix and fiber in the effective configuration are given by the following two relations respectively: (Figure 2.4)

$$\bar{\sigma}_{ij}^m = \bar{B}_{ijkl}^m \bar{\sigma}_{kl} \tag{2.34}$$

and

$$\bar{\sigma}_{ij}^f = \bar{B}_{ijkl}^f \bar{\sigma}_{kl} \tag{2.35}$$

where $\bar{\boldsymbol{B}}^m$ and $\bar{\boldsymbol{B}}^f$ are the undamage stress concentration tensors for the matrix and the fiber, respectively. The experssions of the undamaged stress concentration tensors using the Mori-Tanaka Method are given in the next section. Making use of equations (2.1) and (2.5) in expression (2.34), one obtains

$$\tilde{\sigma}_{ij}^m = (M_{ijpq}^{-m} \bar{B}_{pqrs}^m \tilde{M}_{rskl}) \tilde{\sigma}_{kl} \tag{2.36}$$

or

$$\tilde{\sigma}_{ij}^m = \tilde{B}_{ijkl}^m \tilde{\sigma}_{kl} \tag{2.37}$$

where

$$\tilde{B}_{ijkl}^m = M_{ijpq}^{-m} \bar{B}_{pqrs}^m \tilde{M}_{rskl} \tag{2.38}$$

In equation (2.37), the tensor $\tilde{\boldsymbol{B}}^m$ is the elastic matrix stress concentration tensor in the partial damaged configuration $\tilde{C}$. Similarly, the corresponding elastic fiber stress concentration tensor $\tilde{\boldsymbol{B}}^f$ in the partial damaged configuration $\tilde{C}$ may be obtained such that

$$\tilde{B}_{ijkl}^f = M_{ijpq}^{-f} \bar{B}_{pqrs}^f \tilde{M}_{rskl} \tag{2.39}$$

The variation of the partial damaged stress concentration tensors $\tilde{\boldsymbol{B}}^m$ and $\tilde{\boldsymbol{B}}^f$ with damage is indirectly demonstrated through Figures (2.5) to (2.12).

The material properties are shown in Table 1.1. This is demonstrated for a single lamina loaded axially along the fiber direction. Figure (2.5) shows the variation of the ratio of the axial stress in the fiber to the axial stress in the matrix with respect to the axial fiber damage ϕ_{11}^f in conjunction with several matrix damage cases. Similarly, the variation of the ratio of the axial stress in the matrix to the axial stress in the fiber with respect to the axial damage in conjunction with several fiber damage cases is shown in Figure (2.6). It is clear that the stress ratio is constant in the case when the damage in the matrix is equal to the damage in the fibers. In Figure (2.7) the ratio between the local phase stress and the overall stress is plotted with respect to the fiber damage (i.e., $\phi_{ij}^m = 0$) for several fiber volume fractions,

35

and versus (i.e., $\phi^f_{ij} = 0$) in Figure (2.8). A nonlinear relation is observed in both Figures (2.7) and (2.8). In Figures (2.9) to (2.12), different stress ratios corresponding to those in Figures (2.5), (2.6), (2.7), and (2.8) are plotted with respect to damage ϕ^f_{12} or ϕ^f_{12} .

One assumes a similar relation for strains as that postulated for stresses given, by equation (2.6) such that in the effective configurations $\bar{C}^m$, $\bar{C}^f$ and $\bar{C}$ one obtains

$$\bar{\epsilon}_{ij} = \bar{c}^m \bar{\epsilon}^m_{ij} + \bar{c}^f \bar{\epsilon}^f_{ij} \tag{2.40}$$

where $\bar{\epsilon}^m$ and $\bar{\epsilon}^f$ are the effective matrix and fiber strain tensors, respectively, and $\bar{\epsilon}$ is the effective overall composite strain tensor. In the case of the effective elastic strain concentration factors for the matrix and fiber in the effective configuration $\bar{C}$ as shown in Figure (2.4) one obtains the following expressions:

$$\bar{\epsilon}^m_{ij} = \bar{A}^m_{ijkl} \bar{\epsilon}_{kl} \tag{2.41} (43)$$

and

$$\bar{\epsilon}^f_{ij} = \bar{A}^f_{ijkl} \bar{\epsilon}_{kl} \tag{2.42}$$

where $\bar{A}^m$ and $\bar{A}^f$ are the undamaged strain concentration tensors for the matrix and the fiber, respectively. The experssions of the undamaged strain concentration tensors using the Mori-Tanaka Method are given in the next section.

Making use of the following equations relating the effective elastic strains and the corresponding partial effective elastic strains (Voyiadjis and Park, 1995)[7]

$$\bar{\epsilon}'_{ij} = \tilde{M}^{-1}_{ijkl} \tilde{\epsilon}'_{kl} \tag{2.43}$$

$$\bar{\epsilon}^{m'}_{ij} = M^{-m}_{ijkl} \tilde{\epsilon}^{m'}_{kl} \tag{2.44}$$

$$\bar{\epsilon}^{f'}_{ij} = M^{-f}_{ijkl} \tilde{\epsilon}^{f'}_{kl} \tag{2.45}$$

together with equations (2.41) and (2.42) one obtains the partial damaged elastic strain concentration tensors in the partial damaged $\tilde{C}$ configuration . These are given by the following relations:

$$\bar{A}^m_{ijkl} = M^m_{ijpq} \bar{A}^m_{pqrs} \tilde{M}^{-1}_{rskl} \tag{2.46} (48)$$

and

$$\bar{A}^f_{ijkl} = M^f_{ijpq} \bar{A}^f_{pqrs} \tilde{M}^{-1}_{rskl} \tag{2.47} (48)$$

In Figure (2.13), the strain ratio of the axial strain in the fiber to the axial strain in the matrix with respect to both the matrix and fiber damage is shown to be constant. The strain ratio of the transverse strain in the fiber to the transverse strain in the matrix with respect to fiber damage is shown in Figure (2.14). Similarly, the strain ratio of the transverse strains is shown in Figure (2.15) with respect to matrix damage. The ratio between the local phase strain and the overall strain ($\epsilon^r_{22}/\epsilon_{22}$) is plotted with respect to the fiber damage ϕ^m_{22} (i.e., $\phi^f_{22} = 0$) for several fiber volume fractions in Figure (2.16), and versus ϕ^m_{22} (i.e., $\phi^f_{22} = 0$) in Figure (2.17).

2.6 Mori-Tanaka's Elastic Strain and Stress Concentration Tensors

The expressions for the undamaged elastic stress and strain concentration tensors given here are based on the Mori-Tanaka method. In the recent paper by Chen, et al. (1992)[13], the expressions for the elastic strain concentration tensors $\bar{A}^r$ and the elastic stress concentration factors $\bar{B}^r$ are given by

$$\bar{A}^r_{ijkl} = \bar{H}^r_{ijpq} \bar{F}_{pqkl} \tag{2.48}$$

$$\bar{B}^r_{ijkl} = \bar{J}^r_{ijpq} \bar{G}_{pqkl} \tag{2.49}$$

where

$$\bar{F}_{pqkl} = [\bar{c}^m \bar{H}^m_{pqkl} + \bar{c}^f \bar{H}^f_{pqkl}]^{-1} \tag{2.50}$$

$$\bar{G}_{pqkl} = [\bar{c}^m \bar{J}^m_{pqkl} + \bar{c}^f \bar{J}^f_{pqkl}]^{-1} \tag{2.51}$$

The tensors $\bar{H}^r$ and $\bar{J}^r$ are termed the partial concentration factors for strain and stress and are expressed in the following form:

$$\bar{H}^f_{pqkl} = [I_{pqkl} + \bar{P}_{pqrs}(\bar{E}^f_{rskl} - \bar{E}^m_{rskl})]^{-1} \tag{2.52}$$

$$\bar{J}^f_{pqkl} = [I_{pqkl} + \bar{Q}_{pqrs}(\bar{E}^{-f}_{rskl} - \bar{E}^{-m}_{rskl})]^{-1} \tag{2.53}$$

$$\bar{J}^m_{pqkl} = I_{pqkl} \tag{2.54}$$

where $\bar{E}^f$ and $\bar{E}^m$ are the elastic stiffness tensors of the fiber and matrix, respectively. The tensors $\bar{P}$ and $\bar{Q}$ depend only on the shape ofthe inclusion and on the elastic moduli of the surrounding matrix. For example, for an inclusion in the shape of a circular cylinder in isotropic matrix, the tensor $\bar{P}$ written in matrix form (6x6 array) is given by

$$[\bar{P}] = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{a+4b}{8b(a+b)} & \frac{-a}{8b(a+b)} & 0 & 0 & 0 \\ 0 & \frac{-a}{8b(a+b)} & \frac{a+4b}{8b(a+b)} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{2b} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{2+2b}{2b(a+b)} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{2b} \end{bmatrix} \tag{2.55}$$

$$\tag{2.56}$$

where $r = m, f, d$

$$a \; = \; \frac{\bar{E}^m}{3(1 - 2\bar{\lambda}^m)} + \frac{\bar{G}^m}{3} \tag{2.57}$$

$$b \; = \; \frac{\bar{E}^m}{2(1 - \bar{\lambda}^m)} \tag{2.58}$$

where $\bar{E}^m$ is the Young's modulus of the matrix, $\bar{\lambda}^m$ is the Poisson ratio of the matrix, and $\bar{G}^m$ is the shear modulus of the matrix. The tensor $\bar{Q}$ in equation (2.53) is given by

$$\bar{Q}_{ijkl} \; = \; \bar{E}^M_{ijkl} - \bar{E}^m_{ijpq} \bar{P}_{pqrs} \bar{E}^m_{rskl} \tag{2.59}$$

2.7 Damaged Stress and Strain Concentration tensors Including Interfacial Damage

In order to include the interfacial damage in the damaged stress concentration tensors, two additional damage effect tensors M^{dm} and M^{df} are introduced as shown in Figure (2.18). The M^{dm} termed interfacial damage effect tensor for the matrix is defined as follows:

$$\tilde{\sigma}^m_{ij} \; = \; M^{dm}_{ijkl} \sigma^m_{kl} \tag{2.60}$$

Similiarly, the M^{df} termed interfacial damage effect tensor for the fiber is defined in the same manner as above:

$$\tilde{\sigma}^f_{ij} \; = \; M^{df}_{ijkl} \sigma^f_{kl} \tag{2.61}$$

The overall effective composite stress in the partial effective configuration $\tilde{C}$ is postulated in the same manner as equation (2.6):

$$\tilde{\sigma}_{ij} \; = \; \tilde{c}^m \tilde{\sigma}^m_{ij} + \tilde{c}^f \tilde{\sigma}^f_{ij} \tag{2.62}$$

Similiar to equations (2.7) and (2.8), the stresses of the matrix and fiber in the damaged configuration C are related to the overall stress of the composite by the the full damaged stress concentration tensors such that (Figure (2.18))

$$\sigma^m_{ij} \; = \; B^m_{ijkl} \sigma_{kl} \tag{2.63}$$

and

$$\sigma^f_{ij} \; = \; B^f_{ijkl} \sigma_{kl} \tag{2.64}$$

where B^m and B^f are the damaged stress concentration tensors including the interfacial damage. Substituting equations (2.60)) and (2.61) into (2.62), one obtains the following expression:

$$\tilde{\sigma}_{ij} \; = \; (\tilde{c}^m M^{dm}_{ijkl} B^m_{klpq} + \tilde{c}^f M^{df}_{ijkl} B^f_{klpq}) \sigma_{pq} \tag{2.65}$$

38

Comparing equation (2.3) with (2.65), one obtains the following relation:

$$M_{ijkl}^{d} = (\tilde{c}^{m} M_{ijpq}^{dm} B_{pqkl}^{m} + \tilde{c}^{f} M_{ijpq}^{df} B_{pqkl}^{f}) \tag{2.66}$$

An interfacial damage variable, ϕ^{d}, for the interfacial damage effect tensor M^{d} is defined by Voyiadjis and Park(1995)[7], however the damage variables for the damage effect tensors M^{dm} and M^{df} are not defined directly.

Finally, the damaged stress concentration tensors including the interfacial damage are obtained by making use of equations (2.3), (2.60), (2.61), (2.63) and (2.66) such that

$$B_{ijkl}^{m} = M_{ijpq}^{-dm} \tilde{B}_{pqrs}^{m} M_{rskl}^{d} \tag{2.67}$$

$$B_{ijkl}^{f} = M_{ijpq}^{-df} \tilde{B}_{pqrs}^{f} M_{rskl}^{d} \tag{2.68}$$

Similarly, the damaged strain concentration tensors including interfacial damage are obtained such that

$$A_{ijkl}^{m} = M_{ijpq}^{dm} \tilde{A}_{pqrs}^{m} M_{rskl}^{-d} \tag{2.69}$$

and

$$A_{ijkl}^{f} = M_{ijpq}^{df} \tilde{A}_{pqrs}^{f} M_{rskl}^{-d} \tag{2.70}$$

2.8 Damaged Plastic Stress and Strain Concentration Tensors

In the case when the composite material has undergone plastic deformations, the corresponding expressions for the effective stress concentration tensors for the matrix and the fiber in the configuration are given by the following relations respectively:

$$d\bar{\sigma}_{ij}^{m} = \bar{B}_{ijkl}^{mp} \bar{\sigma}_{kl} \tag{2.71}$$

and

$$d\bar{\sigma}_{ij}^{f} = \bar{B}_{ijkl}^{fp} \bar{\sigma}_{kl} \tag{2.72}$$

where $\bar{B}^{mp}$ and $\bar{B}^{fp}$ are the effective instantaneous plastic stress concentration tensors. These stress concentration tensors are obtained in the same way as their elastic counterparts. The resulting expressions for the partial-damaged plastic stress concentration tensors are expressed as follows:

$$\tilde{B}_{ijkl}^{mp} = m_{ijrs}^{-m} \bar{B}_{rsuv}^{mp} \tilde{m}_{uvkl} \tag{2.73}$$

and

$$\tilde{B}_{ijkl}^{fp} = m_{ijrs}^{-f} \bar{B}_{rsuv}^{fp} \tilde{m}_{uvkl} \tag{2.74}$$

The effective instantneous plastic strain concentration tensors can be determined by making use of the following relations

$$d\bar{\epsilon}_{ij}^m = \bar{A}_{ijkl}^{mp} d\bar{\epsilon}_{kl} \tag{2.75}$$

and

$$d\bar{\epsilon}_{ij}^f = \bar{A}_{ijkl}^{fp} d\bar{\epsilon}_{kl} \tag{2.76}$$

Similarly, the partial-damaged plastic strain concentration tensors are expressed as follows:

$$\tilde{A}_{ijkl}^{mp} = m_{ijrs}^m \bar{A}_{rsuv}^{mp} \tilde{m}_{uvkl}^{-1} \tag{2.77}$$

and

$$\tilde{A}_{ijkl}^{fp} = m_{ijrs}^f \bar{A}_{rsuv}^{fp} \tilde{m}_{uvkl}^{-1} \tag{2.78}$$

Finally, the damaged plastic stress concentration tensors including the interfacial damage are obtained similar to those of equations (2.67) and (2.68) such that

$$B_{ijkl}^{mp} = m_{ijrs}^{-dm} \tilde{B}_{rsuv}^{mp} m_{uvkl}^d \tag{2.79}$$

and

$$B_{ijkl}^{fp} = m_{ijrs}^{-df} \tilde{B}_{rsuv}^{fp} m_{uvkl}^d \tag{2.80}$$

The damaged plastic strain concentration tensors including interfacial damage are obtained such that

$$A_{ijkl}^{mp} = m_{ijrs}^{dm} \tilde{A}_{rsuv}^{mp} m_{uvkl}^{-d} \tag{2.81}$$

and

$$A_{ijkl}^{fp} = m_{ijrs}^{df} \tilde{A}_{rsuv}^{fp} m_{uvkl}^{-d} \tag{2.82}$$

where $\boldsymbol{m}^{dm}$ and $\boldsymbol{m}^{df}$ are the incremental interfacial damage effect tensors for the matrix and the fiber, respectively, while $\boldsymbol{m}^d$ is the incremental interfacial damage effect tensor.

2.9   Conclusion

The stress and strain concentration tensors derived here are for fibrous composites with continuous fibers that undergo damage in both the constituents and the interfacial damage. The damage in the matrix includes nucleation and growth of voids, micro-fracture, etc., while in the fibers includes micro-fracture, etc. In addition, interfacial damage between the matrix and fiber is descibed as debonding damage. The damage stress and strain concentration tensors are obtained for the elasto-plastic states of the material and are based on the Mori-Tanaka method in the undamaged configuration of the material. The derived concentration tensors are functions of the damage effect tensors and undamaged concentration tensors. As a consequence of damage, the volume fractions in the effective undamaged configuration differ from the initial volume fractions. Evolution expressions for the effective volume fractions are also derived in this work. Consistent correlations between stresses, strains, and damage are obtained for the newly derived concentration tensors.
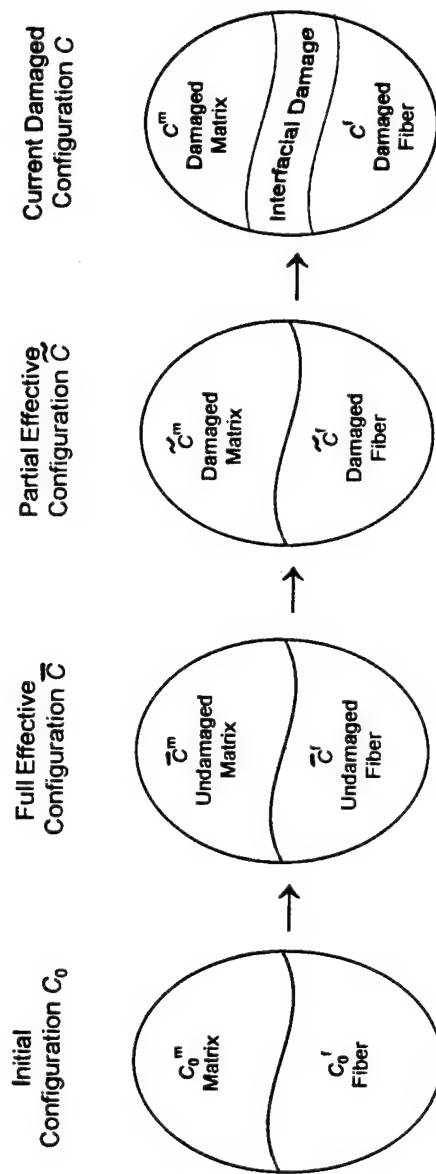
Figure 2.1: Schematic representation of configurations of composites.
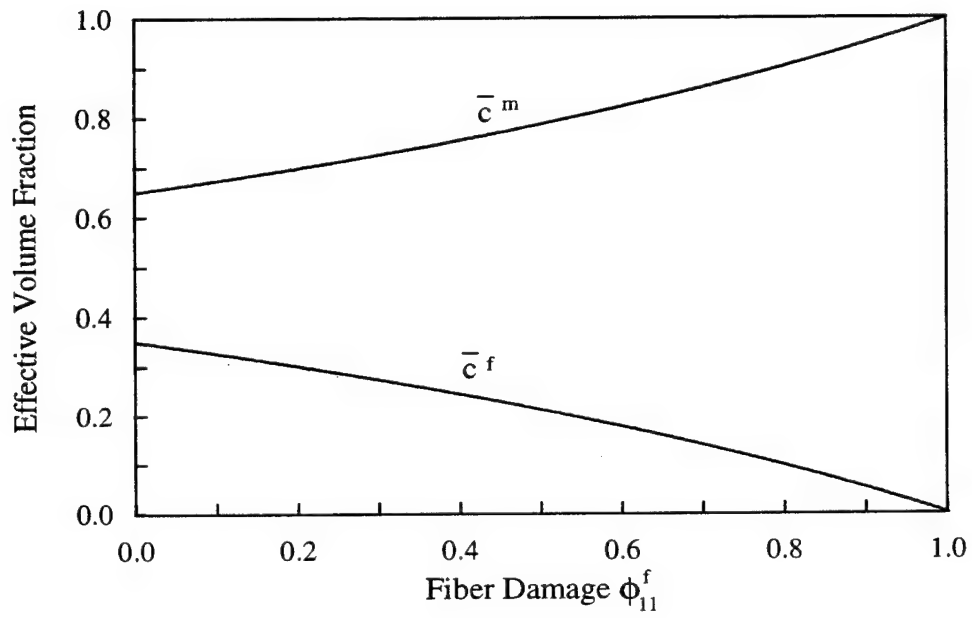
Figure 2.2: Variation of the effective volume fractions with respect to fiber damage.
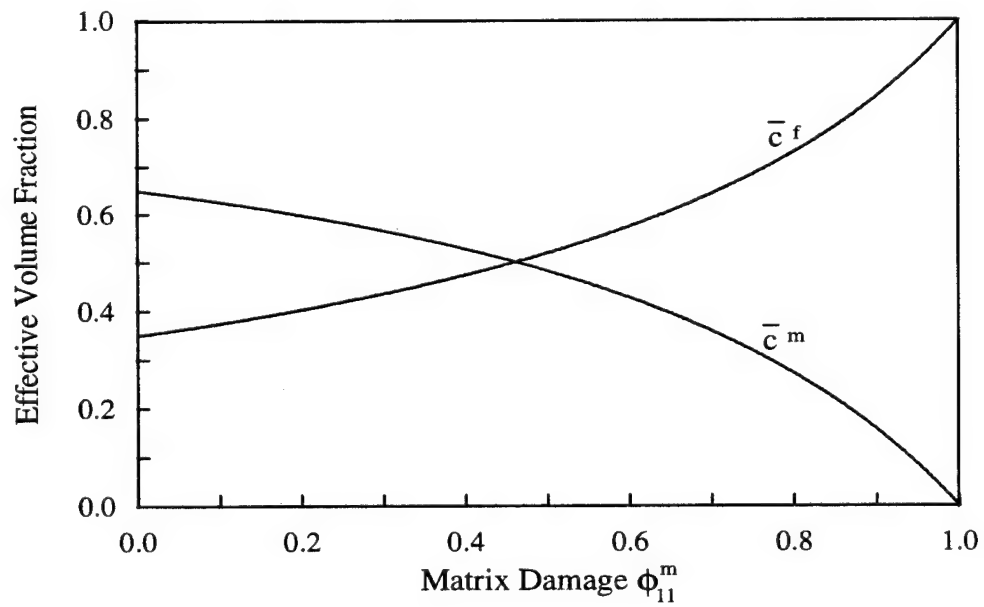
Figure 2.3: Variation of the effective volume fractions with respect to matrix damage.
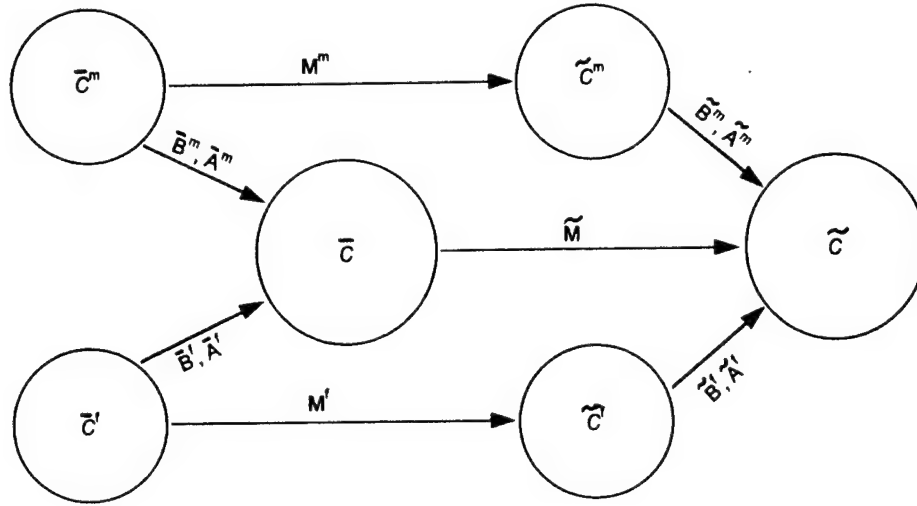
Figure 2.4: Schematic representation of damage in the local constituents of the composite.
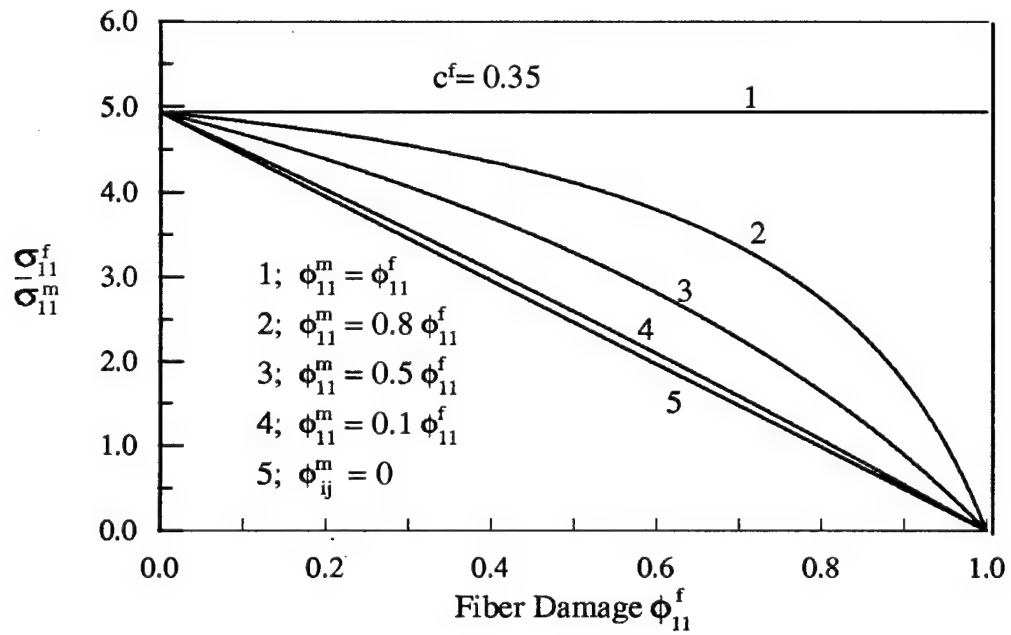
Figure 2.5: Variation of stress ratios of fiber to matrix with respect to fiber damage ϕ_{11}^{f}.
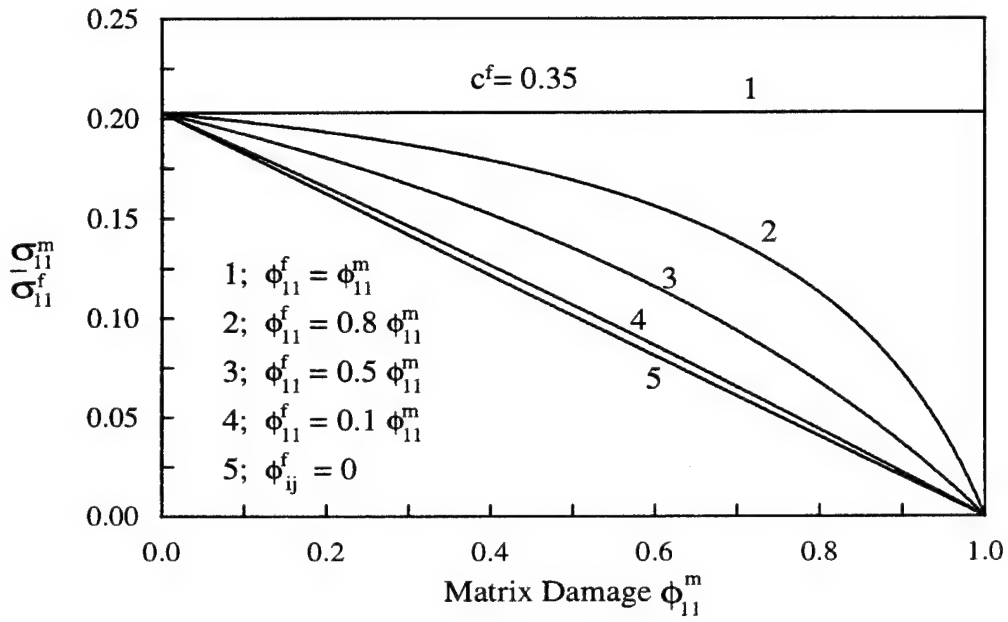
Figure 2.6: Variation of stress ratios of fiber to matrix with respect to fiber damage ϕ_{11}^m.
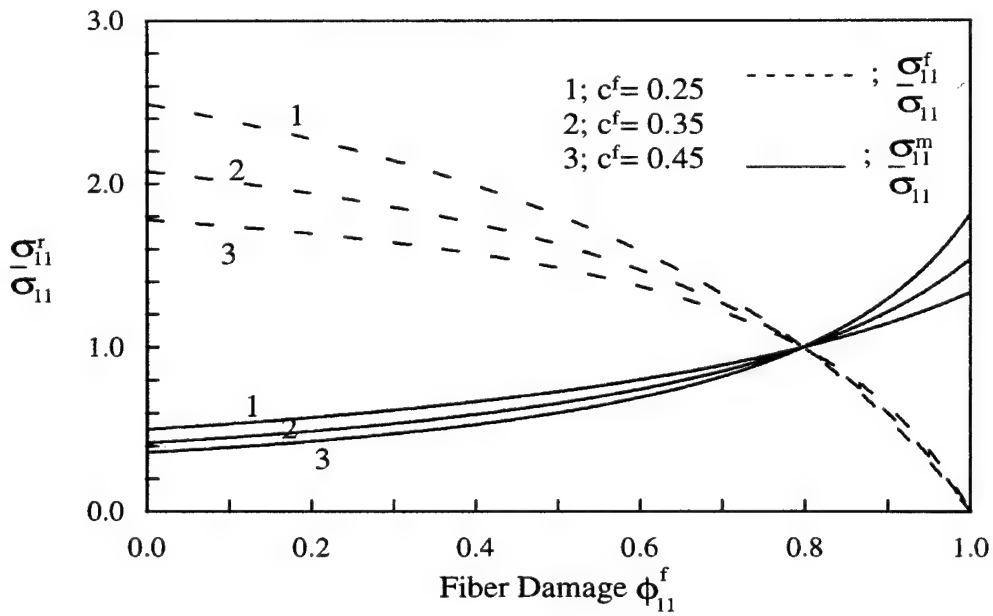
Figure 2.7: Variation of stress ratios of local phases to composite for different fiber fraction with respect to fiber damage ϕ_{11}^f.

44
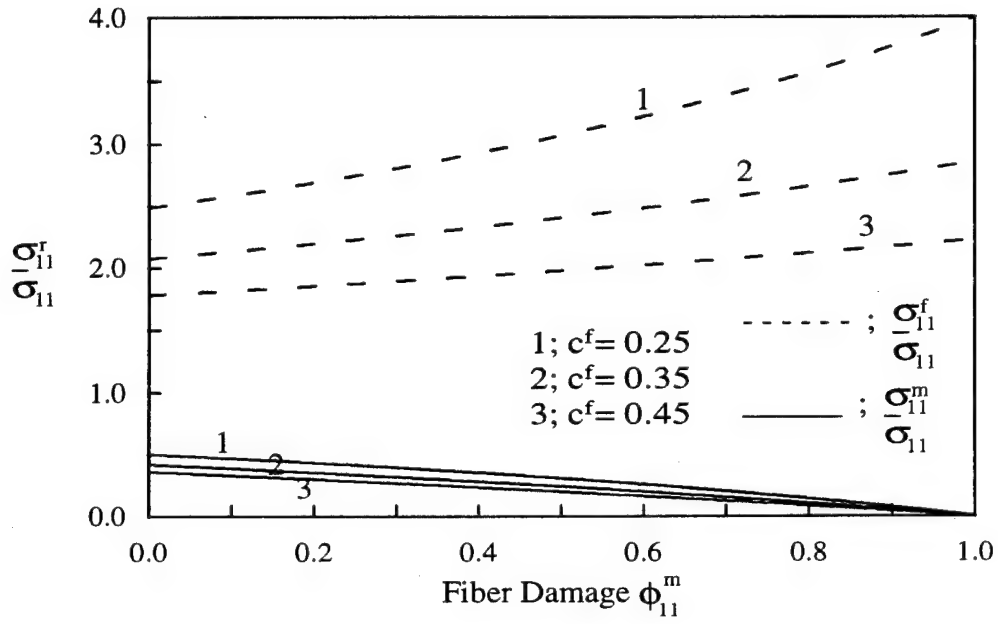
Figure 2.8: Variation of stress ratios of local phases to composite for different fiber fraction with respect to matrix damage ϕ_{11}^m.
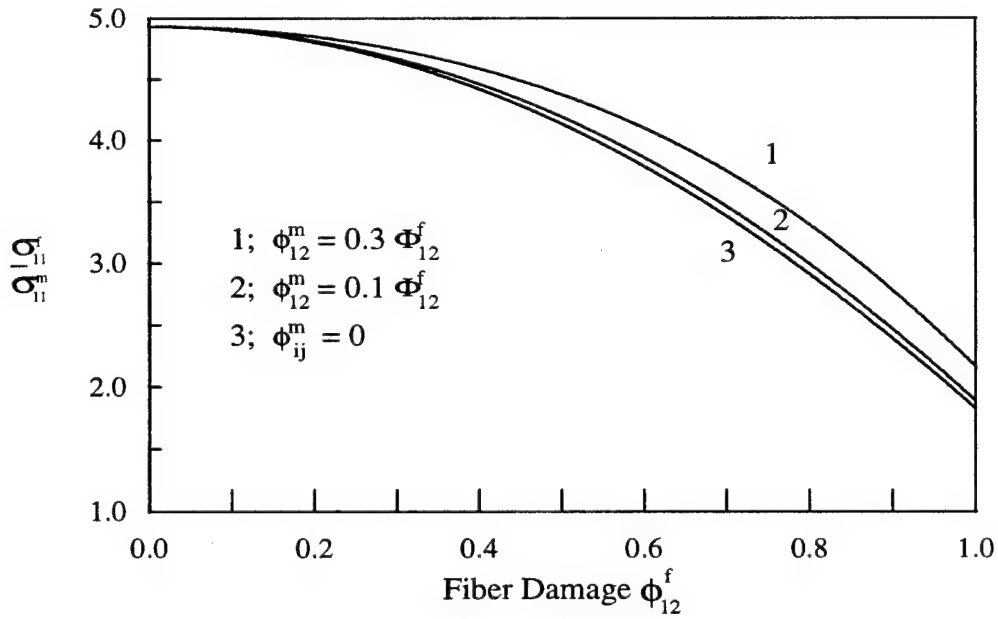
Figure 2.9: Variation of stress ratios of fiber to matrix with respect to fiber damage ϕ_{12}^f.
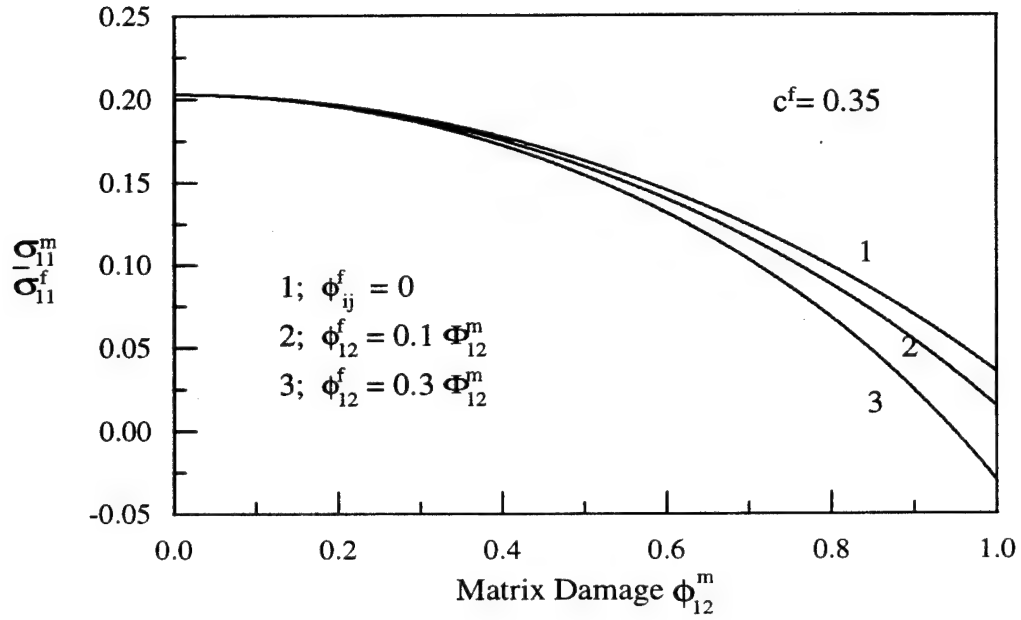
Figure 2.10: Variation of stress ratios of fiber to matrix with respect to matrix damage ϕ_{12}^m.
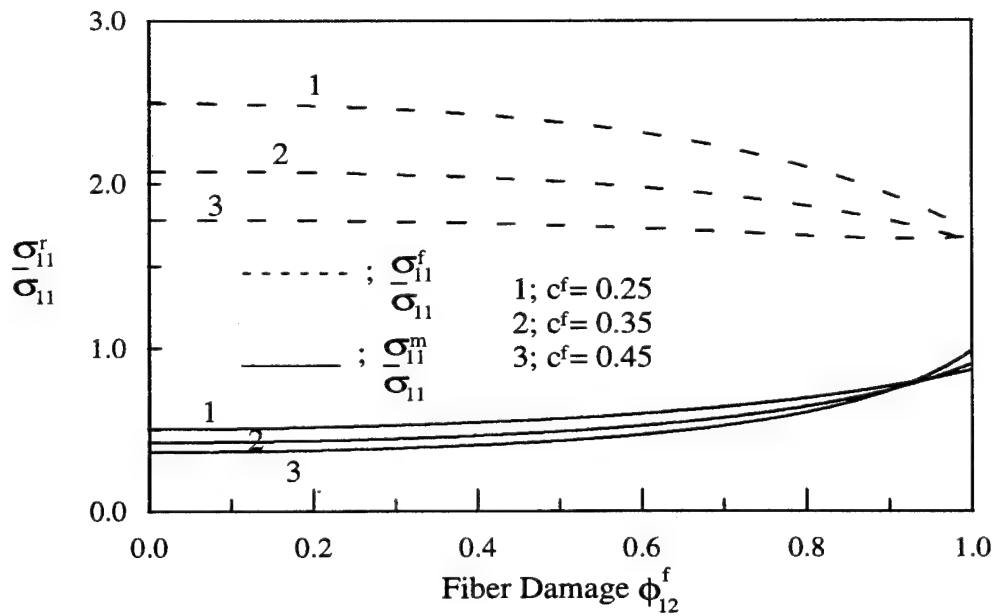
Figure 2.11: Variation of stress ratios of local phases to composite for different fiver volume fractions with respect to fiber damage ϕ_{12}^f.

46
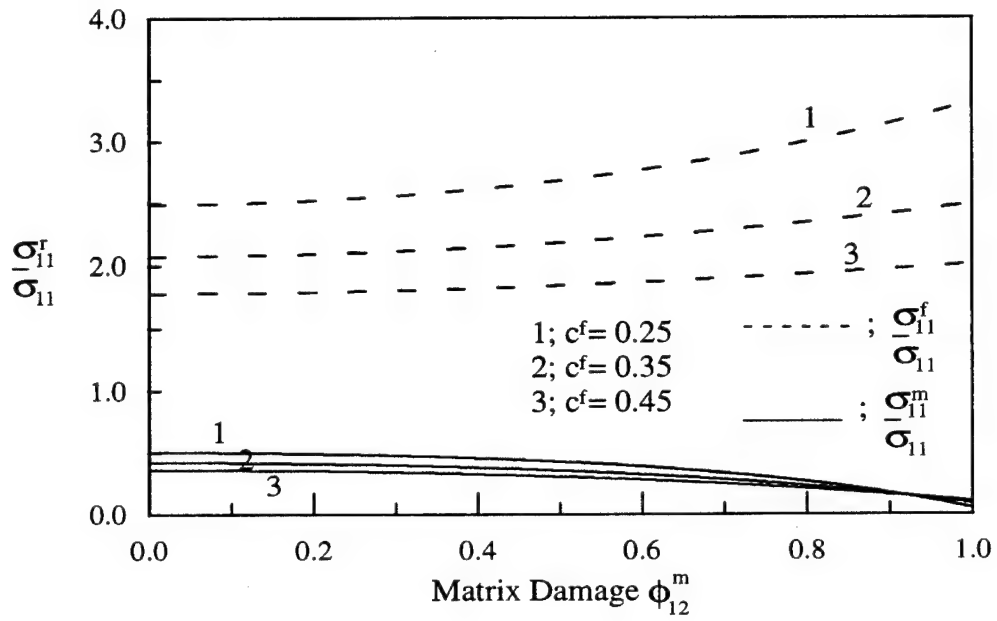
Figure 2.12: Variation of stress ratios of local phases to composite for different fiver volume fractions with respect to fmatrix damage ϕ_{12}^m.
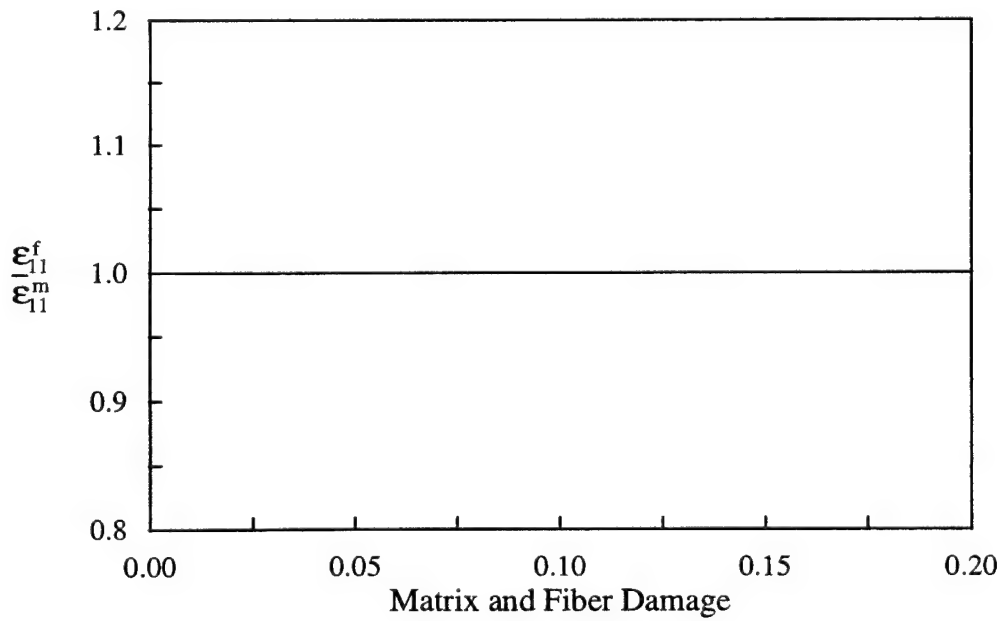
Figure 2.13: Variation of longitudinal strain ratio of fiber to matrix with respect to matrix and fiber damage.
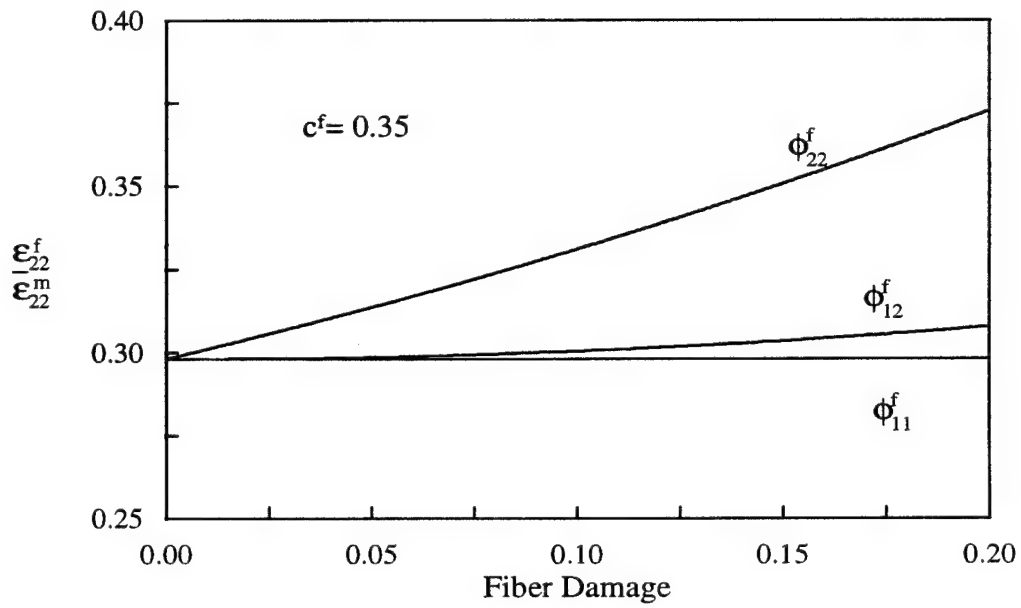
Figure 2.14: Variation of transverse strain ratio of fiber to matrix with respect to fiber damage.
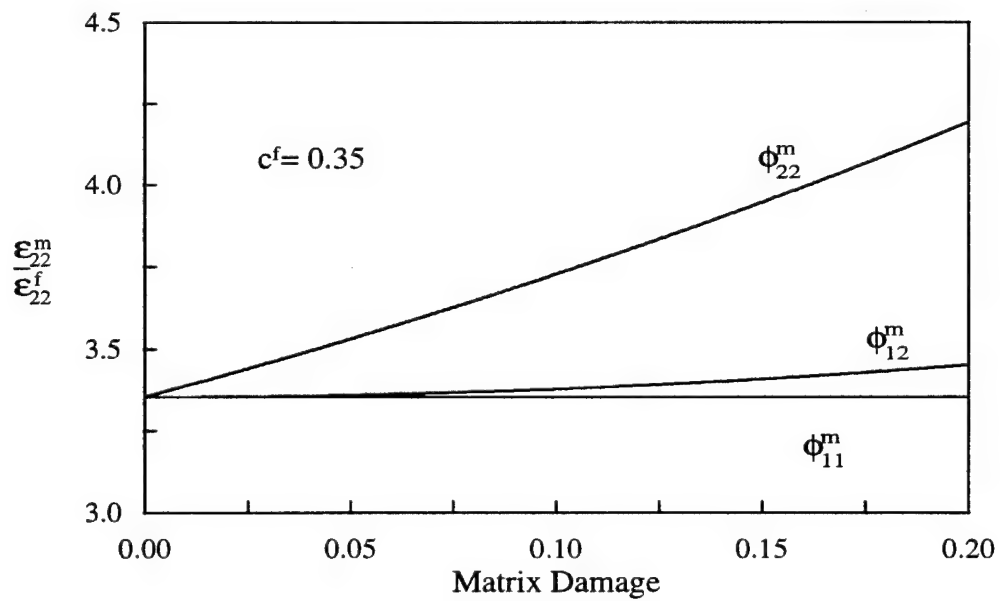
Figure 2.15: Variation of transverse strain ratio of fiber to matrix with respect to matrix damage.
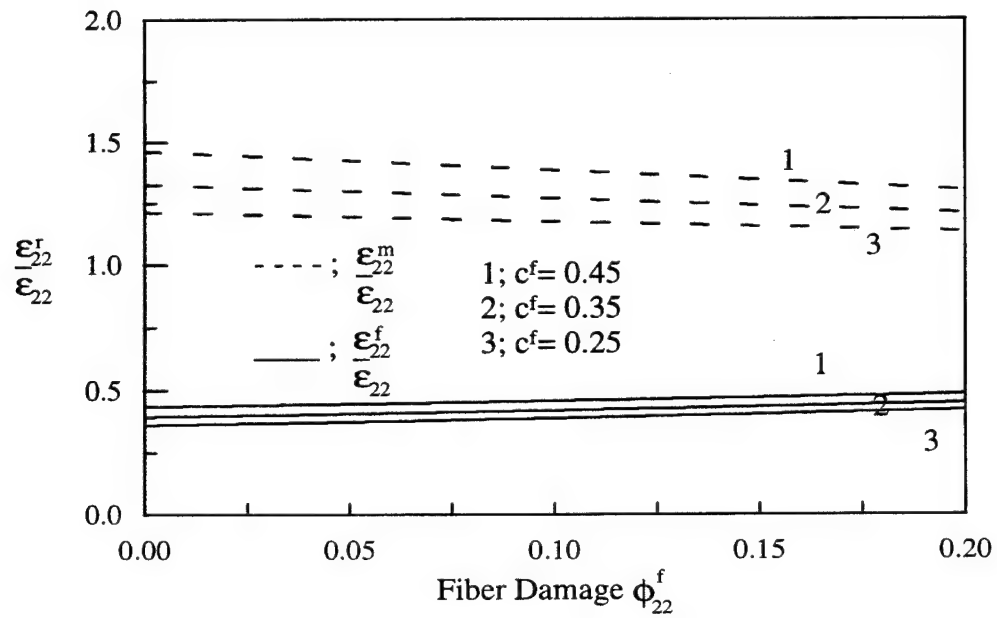
Figure 2.16: Variation of transverse strain ratio of local phases to compositee with respect to fiber damage ϕ_{22}^f.
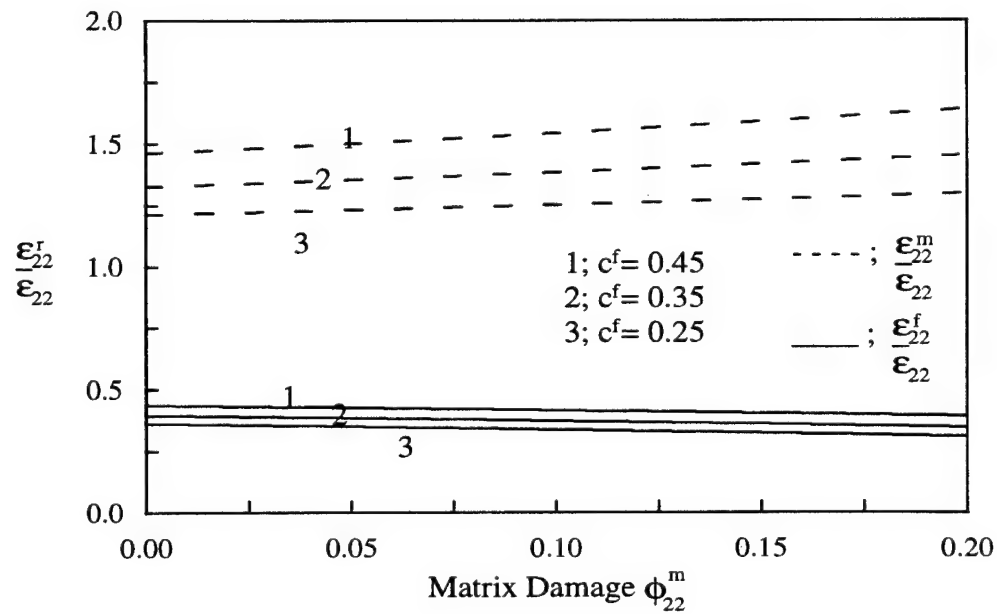
Figure 2.17: Variation of transverse strain ratio of local phases to compositee with respect to matrix damage ϕ_{22}^m.
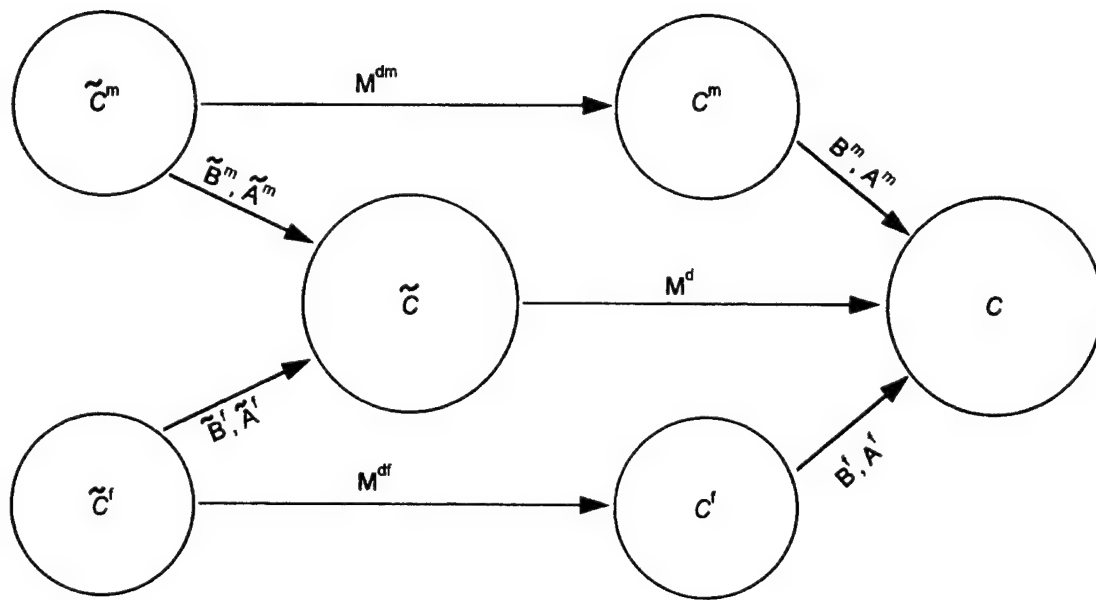
Figure 2.18: Schematic representation of interfacial damage in the composite.

Chapter 3

Local and Interfacial Damage Analysis of Metal Matrix Composites Using the Finite Element Method

3.1   Introduction

A micromechanical damage composite model is used here such that separate local evolution damage relations are used for each of the matrix and the fiber. In addition, this is coupled with interfacial damage between the matrix and the fiber exclusively. An overall response is linked to these damage relations through a certain homogenization procedure. A finite element analysis is used for quantifying each type of damage and predicting the failure loads of dog-bone shaped specimen and center-cracked laminate metal matrix composite plates. The development of damage zones and the stress-strain response are shown for two types of laminated layups, a $(0/90)_s$ layup and a $\pm(45)_s$ layup.

Damage and plastic deformation is incorporated in the proposed model that is used for the analysis of fiber-reinforced metal matrix composite materials. The proposed micromechanical damage composite model used here is such that separate local constitutive damage relations are used for each of the matrix and the fiber. This is coupled with the interfacial damage between the matrix and the fiber exclusively. The damage relations are linked to the overall response through a certain homogenization procedure. Three fourth-order, damage tensors $\boldsymbol{M}^m$, $\boldsymbol{M}^f$ and $\boldsymbol{M}^d$ are used here for the two constituents (matrix and fibers) of the composite system. The matrix damage effect tensor $\boldsymbol{M}^m$ is assumed to reflect all types of damage that the matrix material undergoes such as nucleation and coalescence of voids and microcracks. The fiber damage tensor $\boldsymbol{M}^f$ is considered to reflect all types of fiber damage such as fracture of fibers. An additional tensor $\boldsymbol{M}^d$ is incorporated in the overall formulation that represents interfacial damage between the matrix and the fiber. An overall damage tensor, M, is introduced that accounts for all these separate damage tensors $\boldsymbol{M}^m$, $\boldsymbol{M}^f$, and $\boldsymbol{M}^d$.

3.2 Coordinate Transformation

The three-dimensional damage elasto-plastic constitutive equation for single lamina referring to the principal material coordinate system has been introduced in equation (115). The general three-dimensional constitutive relation of a composite lamina referring to the off-axis coordinate system denoted by prime " ' " can be obtained from equation (1.114) by coordinate transformation. Here, the x-y plane coincides with the $x_1 - x_2$ plane and the angle between the x1 and x axis is θ. The stress and strain vectors in those tow coordinate systems are related by

$$\{d\sigma\} = [T]\{d\sigma\}' \tag{3.1}$$

$$\{d\epsilon\} = [T]\{d\epsilon\}' \tag{3.2}$$

where [T] is a transformation matrix given by

$$[T] = \begin{bmatrix} \cos\theta^2 & \sin\theta^2 & 0 & -2\cos\theta\sin\theta & 0 & 0 \\ \sin\theta^2 & \cos\theta^2 & 0 & 2\cos\theta\sin\theta & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ \cos\theta\sin\theta & \cos\theta\sin\theta & 0 & \cos\theta^2 - \sin\theta^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & \cos\theta & \sin\theta \\ 0 & 0 & 0 & 0 & -\sin\theta & \cos\theta \end{bmatrix} \tag{3.3}$$

Substituting equations (3.1) and (3.2) to (1.114), we obtain the relation

$$\{d\sigma\} = [T]^{-1}[D][T]\{d\epsilon\}' \tag{3.4}$$

Thus, the damage elasto-plastic stiffness matrix referring to the off-axis coordinate $x - y - z$ system is

$$[D]' = [T]^{-1}[D][T] \tag{3.5}$$

The constitutive equation for plane stress problem is obtained from imposing the plane stress conditions $\sigma_{zz} = \sigma_{xz} = \sigma_{yz} = 0$ to equation (3.4). The explicit expression of constitutive equation for plane stress is as follows:

$$\left\{ \begin{array}{c} d\sigma_{xx} \\ d\sigma_{yy} \\ d\sigma_{xy} \end{array} \right\} = \begin{bmatrix} D_{11}^* & D_{12}^* & D_{13}^* \\ D_{21}^* & D_{22}^* & D_{23}^* \\ D_{31}^* & D_{32}^* & D_{33}^* \end{bmatrix} \left\{ \begin{array}{c} d\epsilon_{xx} \\ d\epsilon_{yy} \\ d\epsilon_{xy} \end{array} \right\} \tag{3.6}$$

where

$$
\begin{aligned}
D^*_{11} &= D'_{11} - D'_{13}D'_{31}/D'_{33} \\
D^*_{12} &= D'_{12} - D'_{13}D'_{32}/D'_{33} \\
D^*_{21} &= D'_{21} - D'_{23}D'_{31}/D'_{33} \\
D^*_{13} &= D'_{14} - D'_{13}D'_{36}/D'_{33} \\
D^*_{31} &= D'_{41} - D'_{43}D'_{31}/D'_{33} \\
D^*_{22} &= D'_{22} - D'_{23}D'_{32}/D'_{33} \\
D^*_{23} &= D'_{24} - D'_{23}D'_{34}/D'_{33} \\
D^*_{32} &= D'_{42} - D'_{43}D'_{32}/D'_{33} \\
D^*_{33} &= D'_{44} - D'_{43}D'_{34}/D'_{33}
\end{aligned}
\tag{3.7}
$$

3.3 Gross Damage Elasto-Plastic Stiffness

The elasto-plastic damage stiffness tensor for a single lamina in its principal material coordinate system has been presented in equation (1.115). This stiffness tensor is transformed to the loading coordinate system and expressed as $[D]_k$ in matrix form. A symmetric stacking of plies is considered here such that t is the thickness of the laminate consisting of n plies and t_k is the thickness of the kth lamina. The average stress increment is expressed as follows (in vector form):

$$
\{d\sigma\}_{ave} = \left[\frac{1}{t}\sum_{k=1}^{n}[D^*]_k t_k\right]\{d\epsilon\}
\tag{3.8}
$$

Making use of equation (3.8), one can define the gross damage elasto-plastic stiffness for the laminated composite as follows in matrix form:

$$
[D_g] = \left[\frac{1}{t}\sum_{k=1}^{n}[D^*]_k t_k\right]
\tag{3.9}
$$

Making use of the assumption of constant strain through the laminate thickness, the stresses in each lamina are calculated as follows:

$$
\{d\sigma\}_k = [D^*]_k\{d\epsilon\}
\tag{3.10}
$$

3.4 Finite Element Formulation

The governing equation of the finite element method can be derived from the principle of virtual work such as

$$
\int_V \sigma_{ij}\delta\epsilon_{ij}dV = \int_V q_i\delta u_i dV + \int_A t_i\delta u_i dA
\tag{3.11}
$$

53

where δu_i is a field of virtual displacements that is compatible with applied forces and $\delta \epsilon_{ij}$ is the corresponding field of compatible virtual strains given by

$$\delta \epsilon_{ij} = \frac{1}{2}\left[\frac{\partial(\delta u_i)}{\partial x_j} + \frac{\partial(\delta u_j)}{\partial x_i}\right] \tag{3.12}$$

and q_i and t_i are body forces and surface tractions, respectively. For a small deformation analysis, we have

$$u_i = N_{ij}U_j \tag{3.13}$$

$$\delta u_i = N_{ij}(\delta U_j) \tag{3.14}$$

where U_j is the displacement of nodal points and N_{ij} is the displacement interpolation function or the shape function. Substituting equations (3.12) and (3.14) into (3.11), one obtains the equilibrium equations as follows:

$$\int_V \sigma_{ij}\frac{\partial N_{ia}}{\partial x_j}dV = \int_V q_i N_{ia}dV + \int_A t_i N_{ia}dA \tag{3.15}$$

One finally obtains the incremental equilibrium equations by differentiating both sides of equation (3.15)

$$[K]\{dU\} = \{dP\} \tag{3.16}$$

where $\{dU\}$ is the unknown incremental displacement vector of the nodal points, and $\{dP\}$ is the corresponding incremental nodal forces given by

$$dP_a = \int_V dq_i N_{ia}dV + \int_A dt_i N_{ia}dA \tag{3.17}$$

where dq_i is the incremental body force and dt_i is the incremental surface traction. In equation (3.16), $[K]$ is the stiffness matrix which is given by

$$K_{ab} = \int_V \frac{\partial N_{ia}}{\partial x_j}D_{ijkl}\frac{\partial N_{kb}}{\partial x_l}dV \tag{3.18}$$

The incremental equilibrium equation (3.16) expresses the equilibrium between the internal forces $\{dF\}$ (on the left-hand side) and the external force $\{dP\}$ (on the right-hand side). The residual force vector $\{dR\}$ is defined by

$$\{dR\} = \{dP\} - \{dF\} \tag{3.19}$$

In a damage elastic-plastic analysis, because of the nonlinear relationship between the stress and the strain, the equilibrium equation (3.16) is a nonlinear equation of strains, and therefore, is a nonlinear function of the nodal displacement. Iterative methods are usually employed to solve equation (3.16) for displacements corresponding to a given set of external

loads. Moreover, since a damage elasto-plastic constitutive relation depends on deformation history, an incremental analysis following an actual variation of external forces is used to trace the variation of displacement, strain, stress, and damage along with the external forces.

In an incremental analysis, the total load $\{P\}$ acting on a structure is added in increments step by step. At the $(n+1)$th step, the load can be expressed as

$$^{n+1}\{P\} = {}^{n}\{P\} + {}^{n+1}\{dP\} \tag{3.20}$$

where the left superscript n indicates the nth incremental step. Assuming that the solution at the nth step, are known, and at the $(n+1)$th step, one obtains the following, corresponding to the load increment $\{dP\}$,

$$^{n+1}\{u\} = {}^{n}\{u\} + {}^{n+1}\{du\} \tag{3.21}$$

$$^{n+1}\{\sigma\} = {}^{n}\{\sigma\} + {}^{n+1}\{d\sigma\} \tag{3.22}$$

$$^{n+1}\{\epsilon\} = {}^{n}\{\epsilon\} + {}^{n+1}\{d\epsilon\} \tag{3.23}$$

$$^{n+1}\{\phi^{r}\} = {}^{n}\{\phi^{r}\} + {}^{n+1}\{d\phi^{r}\} \tag{3.24}$$

3.4.1 Solution

A full Newton-Raphson method is used in this work to solve the system of nonlinear equations that arise from the equilibrium equations. A brief description of the method is given by Voyiadjis (1973)[15]. The incremental analysis technique described in this chapter is successfully implemented into the finite element program NDA (Nonlinear Damage Analysis) using the above described iterative method. The steps involved in the process of solving are briefly described below.

- INCREMENT: Loop for each load increment

    a) Calculate the load or applied displacement increment for the current incremental step or input the load/applied displacement increment.

    b) ITERATE: Loop for full Newton-Raphson iteration:

        1) Compute the residual load vector for this iteration subtracting the equilibrium load from the load computed for the increment.

        2) Rotate the appropriate loads and applied displacements such that the degrees of freedom at the skew boundary (a boundary condition that is not along the global coordinate system) are normal and tangential to the skew boundary.

        3) Assemble the stiffness matrices and find the equivalent loads for the applied incremental displacements. Since explicit integration is difficult, Gaussian points are used to evaluate the above integrals.

55

4) Solve for the incremental displacements using a linear solver.

5) Add the solved iterative incremental displacements to the applied incremental displacements to obtain the complete iterative incremental displacements.

6) Rotate back the complete iterative incremental displacements at the skew boundaries to the global coordinate system.

7) Cumulate the complete iterative incremental displacements to the total incremental displacements.

8) Find the stresses due to the iterative incremental displacements. From the iterative deformation gradient and the stresses updated, compute the updated constitutive matrix D. From the total incremental displacements accumulated so far and the D matrix, calculate the equilibrium load vector.

9) Check if the convergence of solution is met using a particular convergence criterion. If convergence has not occurred, go back to the step ITERATE.

c) If divergence occurs according to the convergence criterion, then reduce the load increment appropriately as specified by the user and start the iterative solution over again for that load increment.

d) If divergence occurs for a load increment that has been reduced 'm' times (specified by the user), then report 'convergence not met' and leave the solution phase.

e) If convergence has occurred, then perform the following operations before going for the next increment.

1) Update the nodal positions by adding the currently obtained incremental displacements.

2) Transform the quantities pertaining to the material property to the present configuration.

3) Print out the appropriate quantities pertaining to the converged increment according to the user's specifications.

f) If the total load is not reached, go back to the step INCREMENT.

3.4.2   Stress and Damage Computations

- Step 1. Retrieve sij, srij, frij. Retrieve also the information whether the previous loading was a damage loading or not (IDAMG) and plastic loading or not (IYILD).

a) If IDAMG $= 0$ when retrieved, then evaluate the incremental elastic-predictor stress σ_{ij}^p assuming that the loading is elastic. Use the undamaged elastic stiffness matrix for the calculation ($d\sigma_{ij}^p = \bar{E}_{ijkl}d\epsilon_{kl}$).

b) If IDAMG $\neq 0$ when retrieved, use ($d\sigma_{ij}^p = E_{ijkl}d\epsilon_{kl}$).

c) Calculate the incremental elastic-predictor stress of matrix constituent $d\sigma_{ij}^{mp}$.

d) Check if the predicted stress state of matrix constituent is inside the yield surface or not.

e) If the stress state of matrix constituent is inside the yield surface then:

56

1) Assign elastic stiffness to the constitutive stiffness and the predictor stress increment to the actual computed stress increment.

2) Set IYILD = 0 indicating the elastic loading has taken place.

3) Exit to Step 2. Otherwise, go to the next step.

f) Set IYILD = 1, then:

1) Calculate the elasto-plastic stiffness D (when IDAMG = 1) or $\bar{D}$ (when IDAMG = 0).

2) Update the quantities σ_{ij}, σ_{ij}^m, σ_{ij}^f, α_{ij}^m.

- Step 2

a) Check the damage criteria using the updated quantity σ_{ij}^r.

b) If damage criteria $g^r < 0$, then IDAMG = 0. Exit from the routines.

c) If damage criteria $g^r > 0$, then IDAMG = 1. Calculate the damage increment $d\phi_{ij}^r$ and update damage quantity ϕ_{ij}.

d) Store the updated quantities in a file.

3.5 Application to the Dog-Bone Shaped Specimen and the Center-Cracked Laminated Plates

The finite element method is used for solving a dog-bone shaped specimen and a center-cracked laminate plate shown in Figure 3.1 that is subjected to inplane tension. Due to symmetry in geometry and loading as shown in Figure 3.1, one-quarter of the plate needs to be analyzed.

Two-dimensional plane stress analysis rather than three-dimensional analysis is used here since the thickness of plate is much smaller than the other dimensions. Applying the appropriate boundary conditions for the symmetry, both one-quarter of the center-cracked laminate plate and the dog-bone shaped specimen are discretized using plane stress finite elements. The finite element meshes chosen for analyzing the problems are shown in Figure 3.2.

The four-noded quadrilateral element is used in both finite element analyses. Two types of laminate layups $(\pm 45)_s$ and $(0/90)_s$ each consisting of four plies are used here. The thickness of each ply is equal to 0.254 mm. Since both layups are symmetric, no curvature is assumed. Hence, the strain through the plate thickness is assumed to be the same. The material properties and damage parameters using the proposed constitutive model are listed in Table 1.1 and Table 1.4, respectively.

The following convergence criterion is used in this analysis which is based on the incremental internal energy for each iteration in that incremental loading (Bathe, 1990)[16]. It represents the amount of work done by the out-of-balance loads on the displacement increments. Comparison is made with the initial internal energy increment to determine whether

57
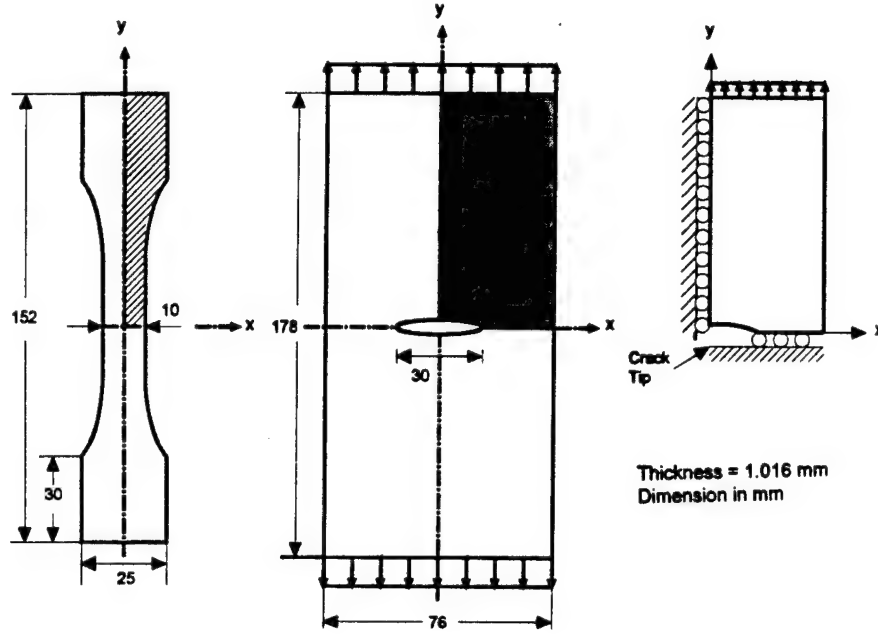
Figure 3.1: Dog-bone shaped specimen and center-cracked laminated plate.

or not convergence has occurred. Convergence is assumed to occur if for an energy tolerance ϵ_E, the following condition is met:

$$\Delta U^{(i)}(^{n+1}R - ^{n+1}F^{(i-1)}) \leq \epsilon_E(\Delta U^{(i)}(^{n+1}R - ^{n}F)) \tag{3.25}$$

where $\Delta U^{(i)}$ is the incremental displacement residual at the (i)th iteration, $(^{n+1}R - ^{n+1}F^{(i-1)})$ is the out-of-balance force vector at (i-1) iteration, and $(\Delta U^{(i)}(^{n+1}R - ^{n}F))$ is the internal energy term for the (i)th iteration in the (n+1)th increment. Divergence is assumed to occur if the out-of-balance internal energy for the (i-1)th iteration is greater than the out-of-balance internal energy for the (i)th iteration.

The load is incremented with uniform load increments of 5 MPa until the principal maximum local damage value ϕ_p^r reaches 1.0 $\phi_p^r \leq 1.0$). The principal maximum local damage value ϕ_p^r is given by:

$$\phi_p^r = \frac{\phi_{11}^r + \phi_{22}^r}{2} + \sqrt{(\frac{\phi_{11}^r - \phi_{22}^r}{2})^2 + \phi_{12}^{r2}} \tag{3.26}$$

Consequently, material failure at integration point is assumed when $\phi_p^r \geq 1$. The principal damage value of the integration point in all elements is monitored at each load increment since it is used to determine the onset of macro-crack initiation of the material. The dog-bone shaped specimen failed when the final load of 270 MPa is reached for the $(\pm 45)_s$ layup and 480 MPa for the $(0/90)_s$ layup. These failure loads are close to the experimental failure loads 276 MPa for the $(\pm 45)_s$ layup and 483 MPa for the $(0/90)_s$ layup (Voyiadjis and Venson, 1995)[10]. The material failure for the center-cracked specimen occurs at the front of the crack tip when the final load of 80 MPa is reached for the $(\pm 45)_s$ layup plate and 120 MPa for the $(0/90)_s$ layup plate.
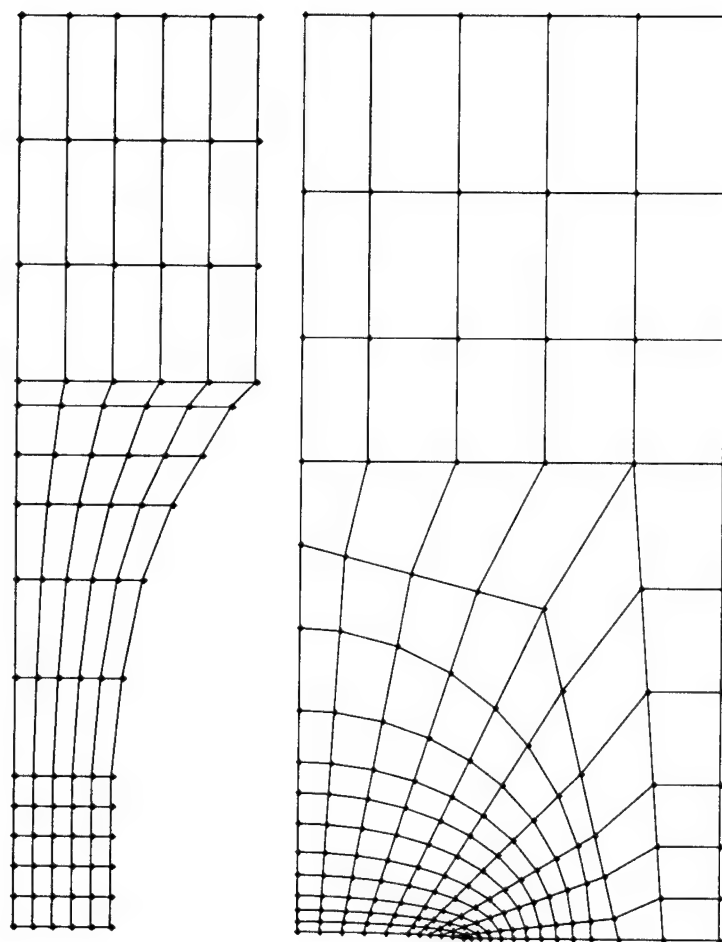
Figure 3.2: Finite element meshes.
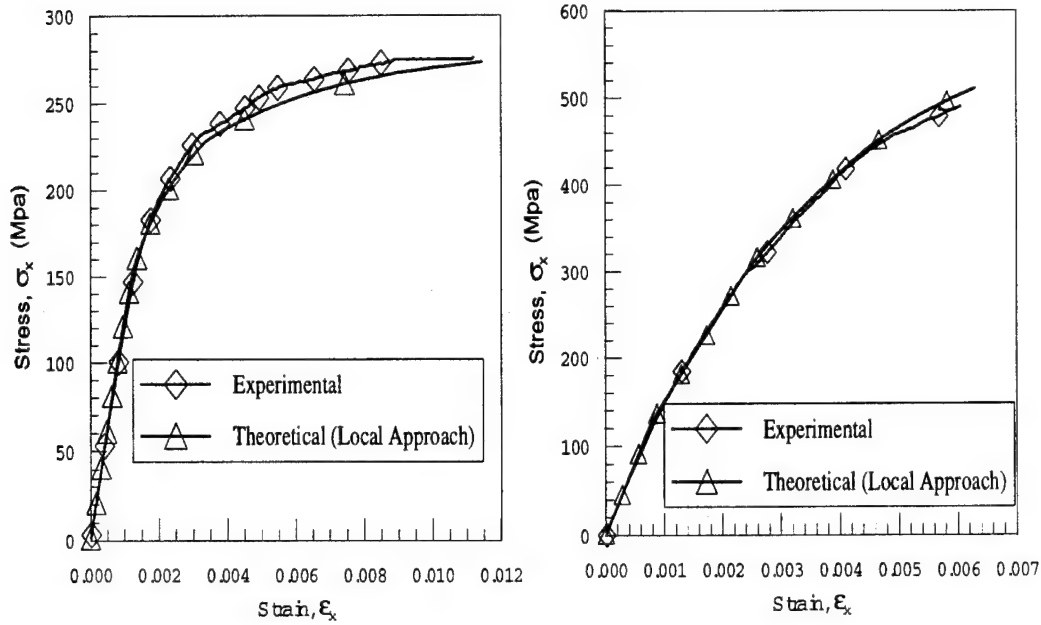
Figure 3.3: Stress-strain curves of $[\pm 45]_s$ layup and $[0/90]_s$ layup

3.6 Discussion of the Results

The stress-strain curves from both the finite element analyses and experiments of the two types of layups of the dog-bone shaped specimens are shown in Figure 3.3. Good correlation is shown between the finite element analysis results and the experimental data obtained by Voyiadjis and Venson (1995)[10]. Strain contours for the $(\pm 45)_s$ layup and $(0/90)_s$ layup of the center cracked plates are shown in Figures (3.4) and (3.5), respectively. Since the two types of layups are symmetric, the strains in each laminae of the layup are the same. However, the stress and damage distributions are different for each laminae of the layup since each laminae has a different stiffness. Stress contours for each laminae are indicated in Figure (3.6) for the $(\pm 45)_s$ layup and Figure (3.7) for the $(0/90)_s$ layup. In Figures (3.8) and (3.9), comparison is made between the damage analysis and the elastic analysis for the stress syy contours around the crack tip. The damage analysis shows considerable stress reduction due to the damage around the crack tip. The stress σ_{yy} at the front of the crack tip as obtained from the elastic solution is higher than that of the material strength of the layup. However, in the damage elasto-plastic analysis, the stresses are reduced such that they are close to those of the material strength. The σ_{yy} stress reductions at the front of the crack tip are more than 50Stress redistributions are clearly indicated in Figures (3.8) and (3.9). Primarily due to the stress reduction around the crack tip, the stress is therefore transferred to the outer portion away from the crack tip. This is clearly indicated in Figure (3.9) where the stress reduction at the 90^0 ply is primarily due to considerable interfacial damage.

The local damage contours around the crack tip are shown in Figures (3.10), (3.11), (3.12), and (3.13), for the failure loads in the case of $[+45]$, $[-45]$,$[0]$, and $[90]$ ply, respectively. For

60

the $[\pm 45]_s$ layups, all types of damages such as matrix, fiber and interfacial are developed. Fiber damage is considerably more spread in the [0] ply than the interfacial damage. On the otherhand, interfacial damage is more pronounced with matrix damage for the [90] ply. However, fiber damage is much less developed in the case of the [90] ply. This is in line with the experimental results obtained by Voyiadjis and Venson (1995)[10].
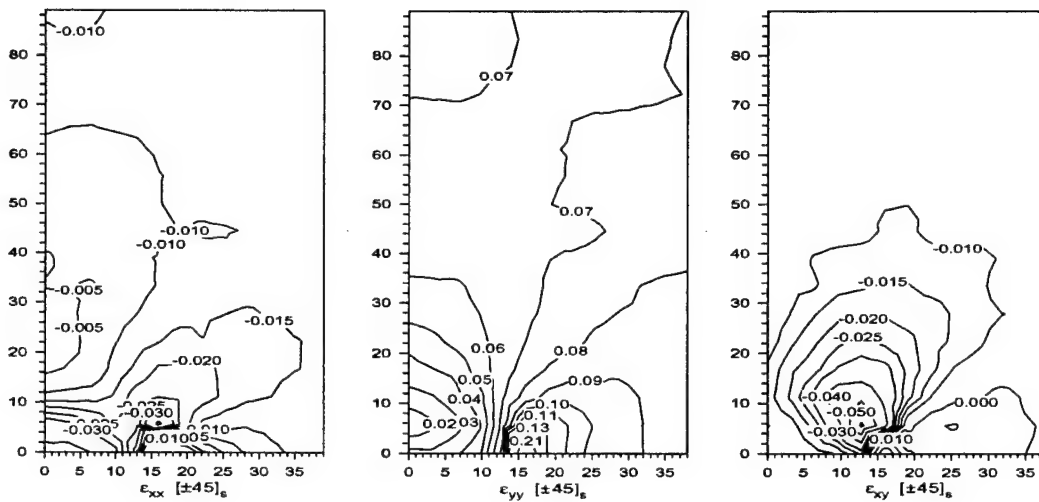
Figure 3.4: Strain contours for $[\pm 45]_s$ layup (in %)

3.7 Summary and Conclusions

The proposed constitutive model is implemented numerically using the finite element method. The model is used to analyze the dog-bone shaped specimens and the center-cracked laminated plates subjected to inplane tensile forces. Very good correlations are demonstrated between the numerical results obtained using the proposed theories and the experimental results for uniaxial tension. The stress and damage contours in the case of the center cracked plate show that stress redistributions and damage are qualitatively in line with the physics of deformation. The analysis presented here allows the separate quantification of the different types of damages such as matrix, fiber or debonding.

The authors are currently working on damage due to delamination which will be introduced into the proposed model in future work. In order to capture delamination due to interlamina stresses, a three-dimensional, finite element analysis will be performed.
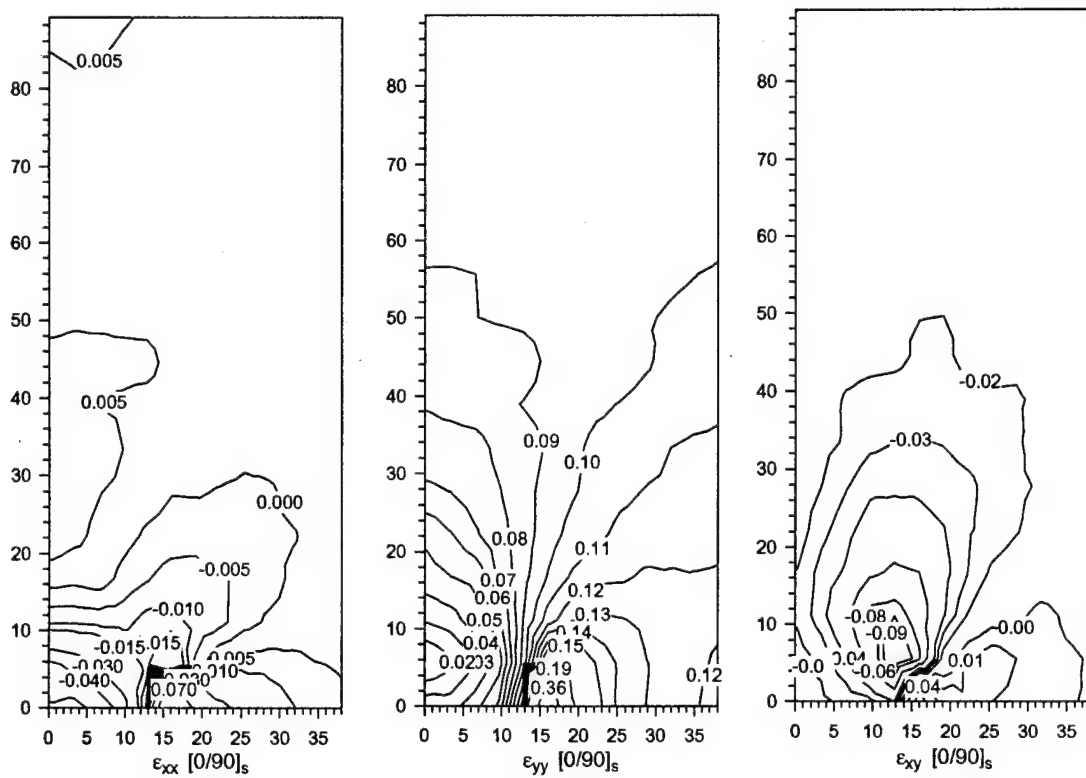
Figure 3.5: Strain contours for $[0/90]_s$ layup (in %)
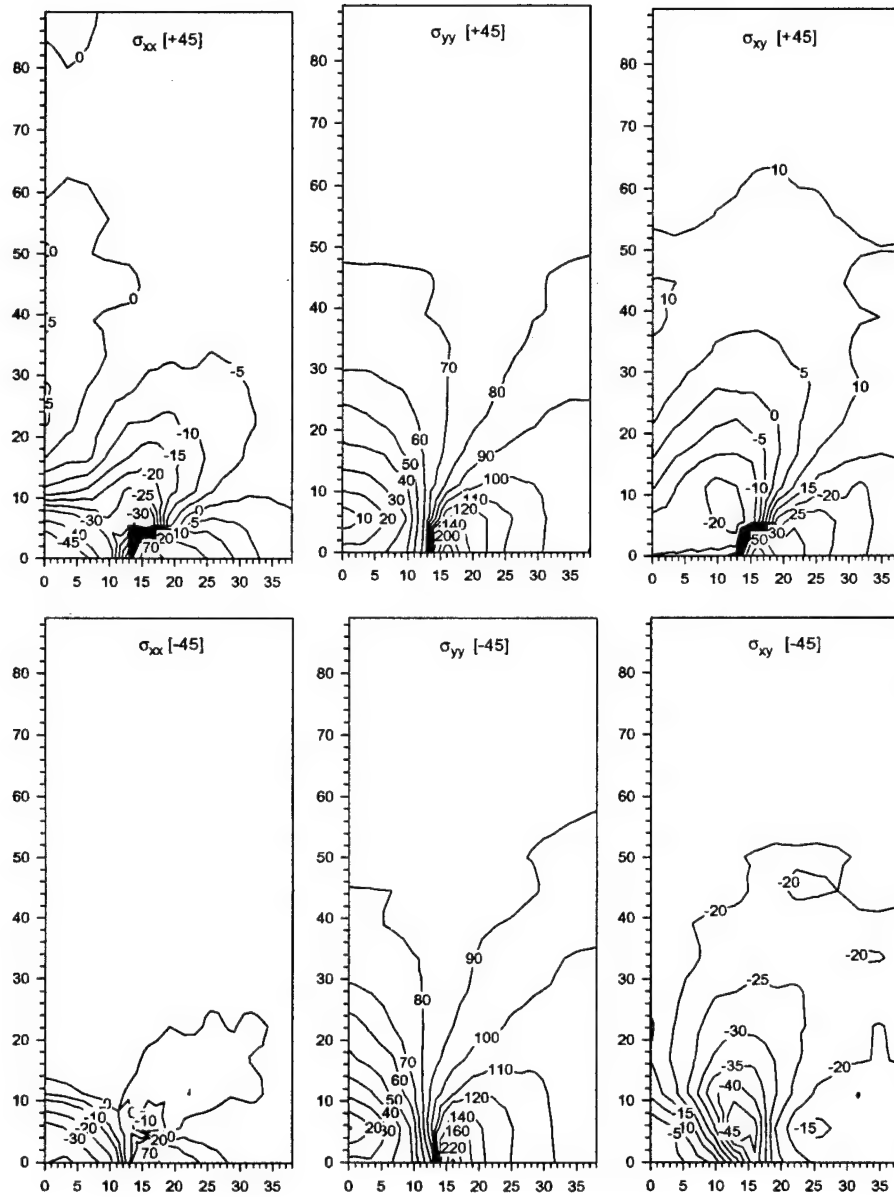
Figure 3.6: Stress contours for $[\pm 45]_s$ layup (units are in MPa).
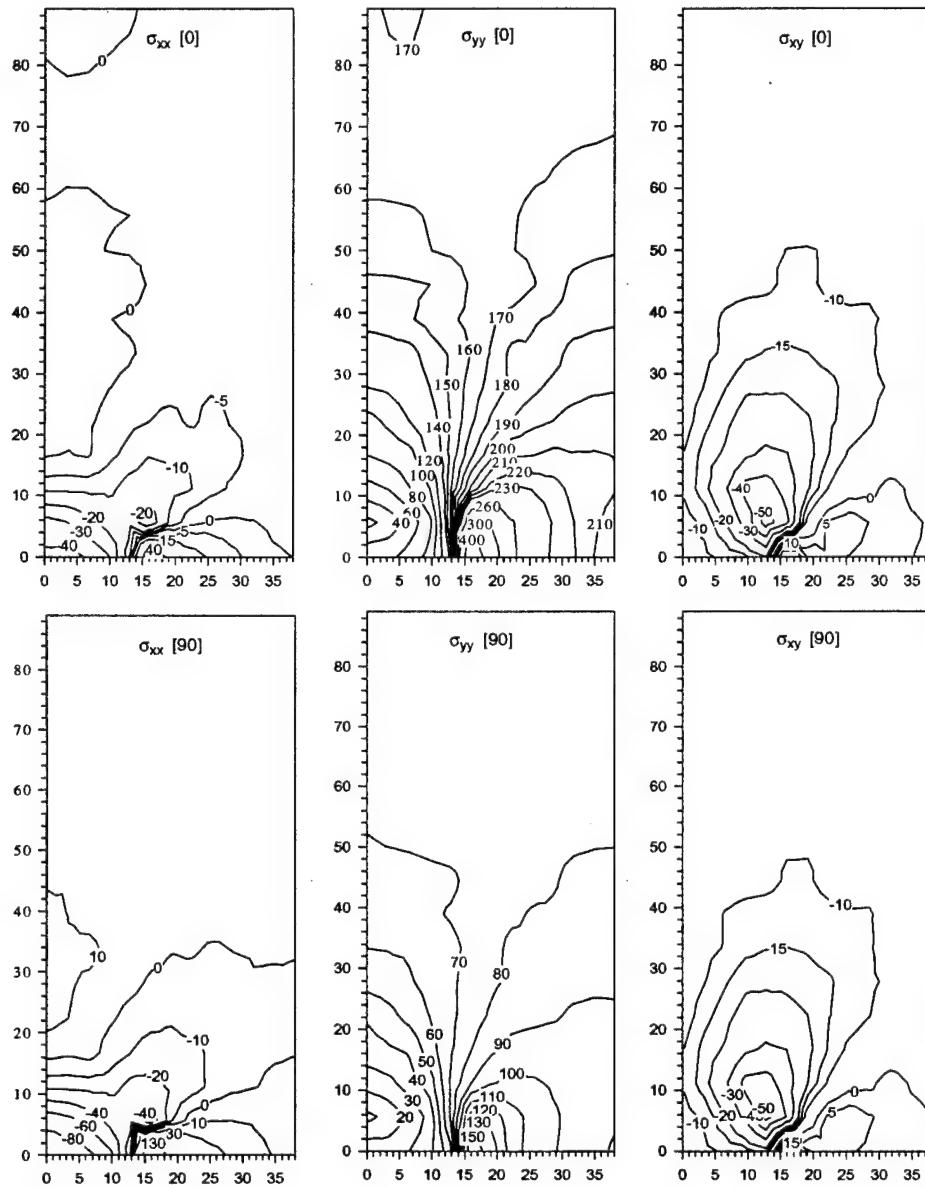
Figure 3.7: Stress contours for $[0/90]_s$ layup (units are in MPa).
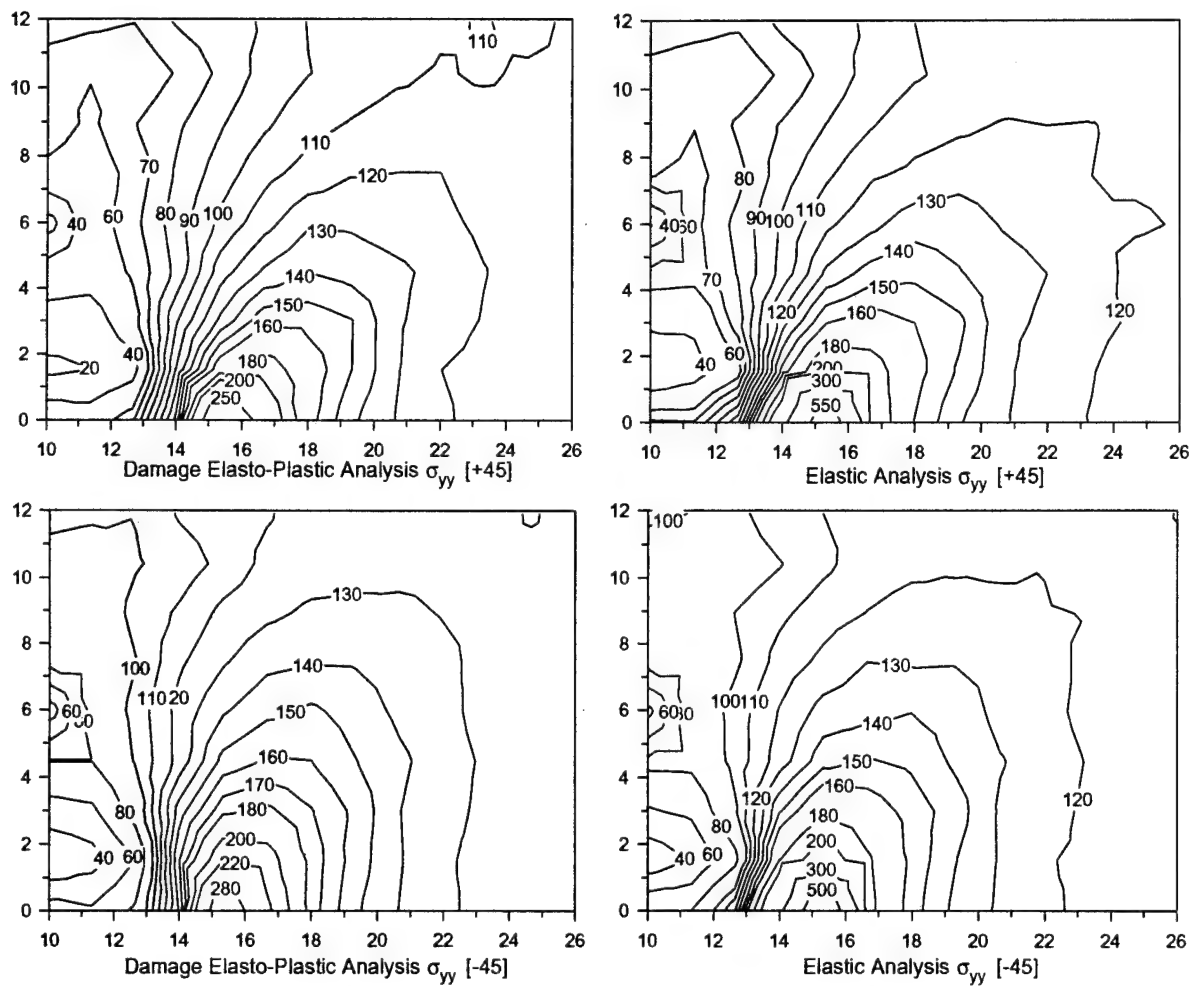
Figure 3.8: Comparison of damage elasto-plastic analysis with elastic analysis of stress σ_{yy} contours around the crack tip for $[\pm 45]_s$ layup (units are in MPa).

65
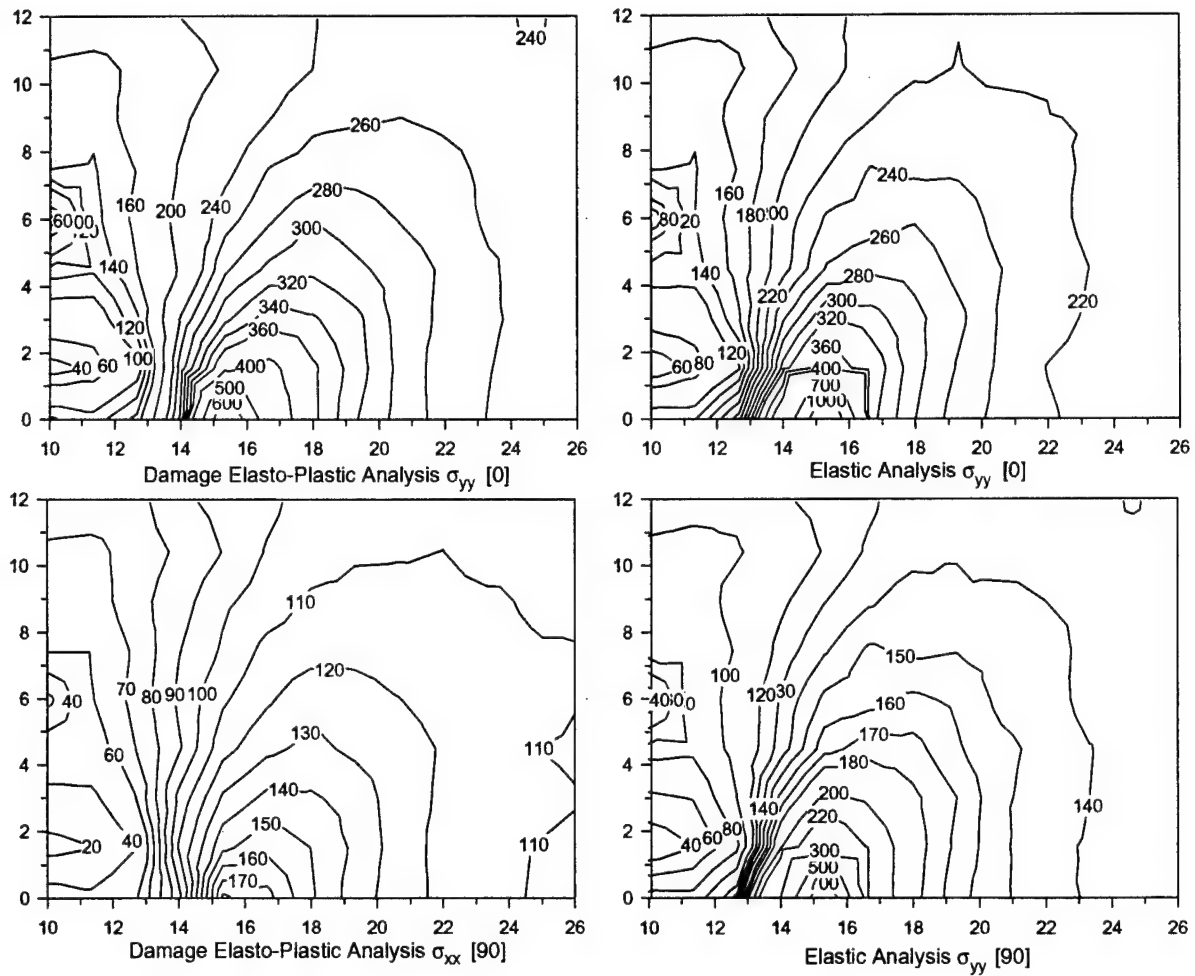
Figure 3.9: Comparison of damage elasto-plastic analysis with elastic analysis of stress σ_{yy} contours around the crack tip for $[0/90]_s$ layup (units are in MPa).

66
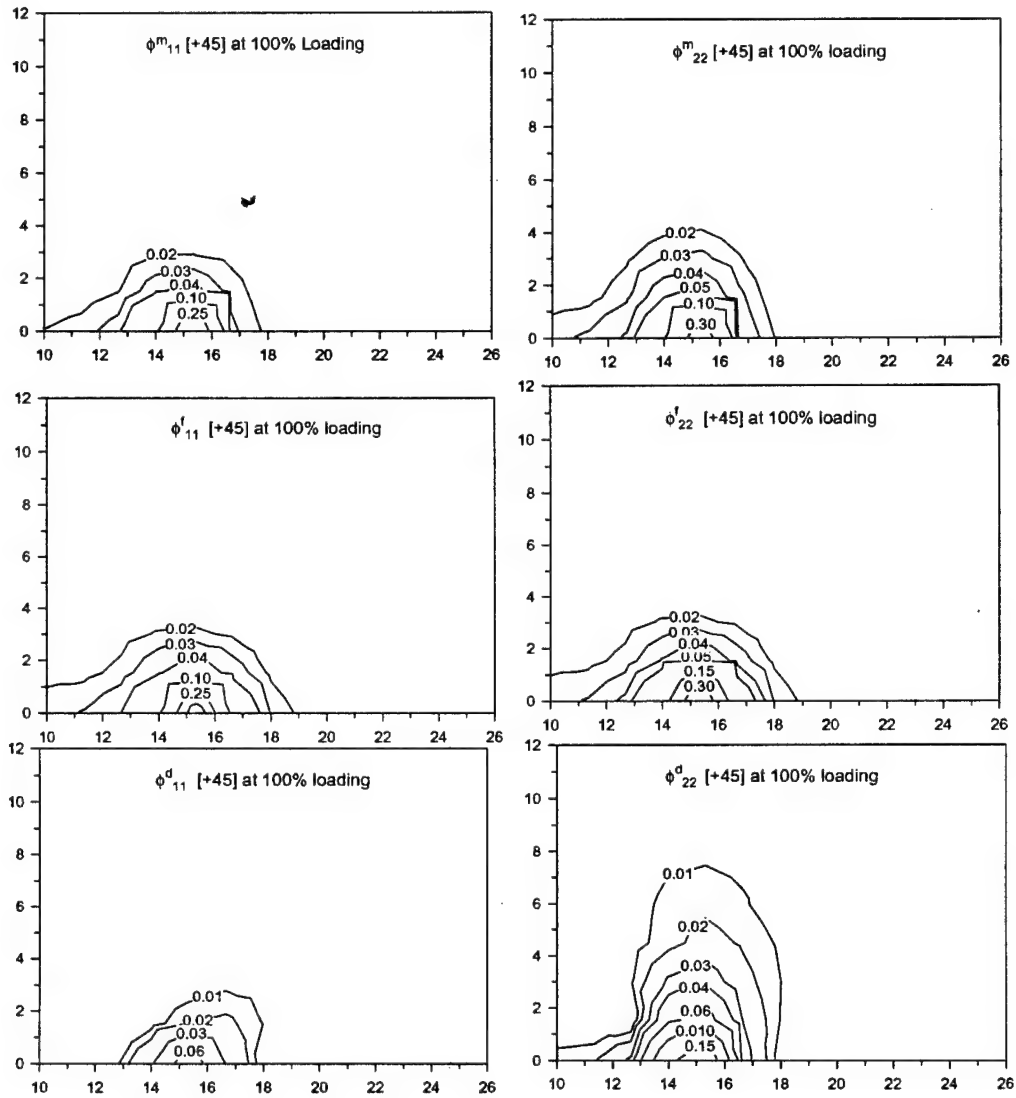
Figure 3.10: Damage contours around crack tip at the failure load for [+45] lamina.
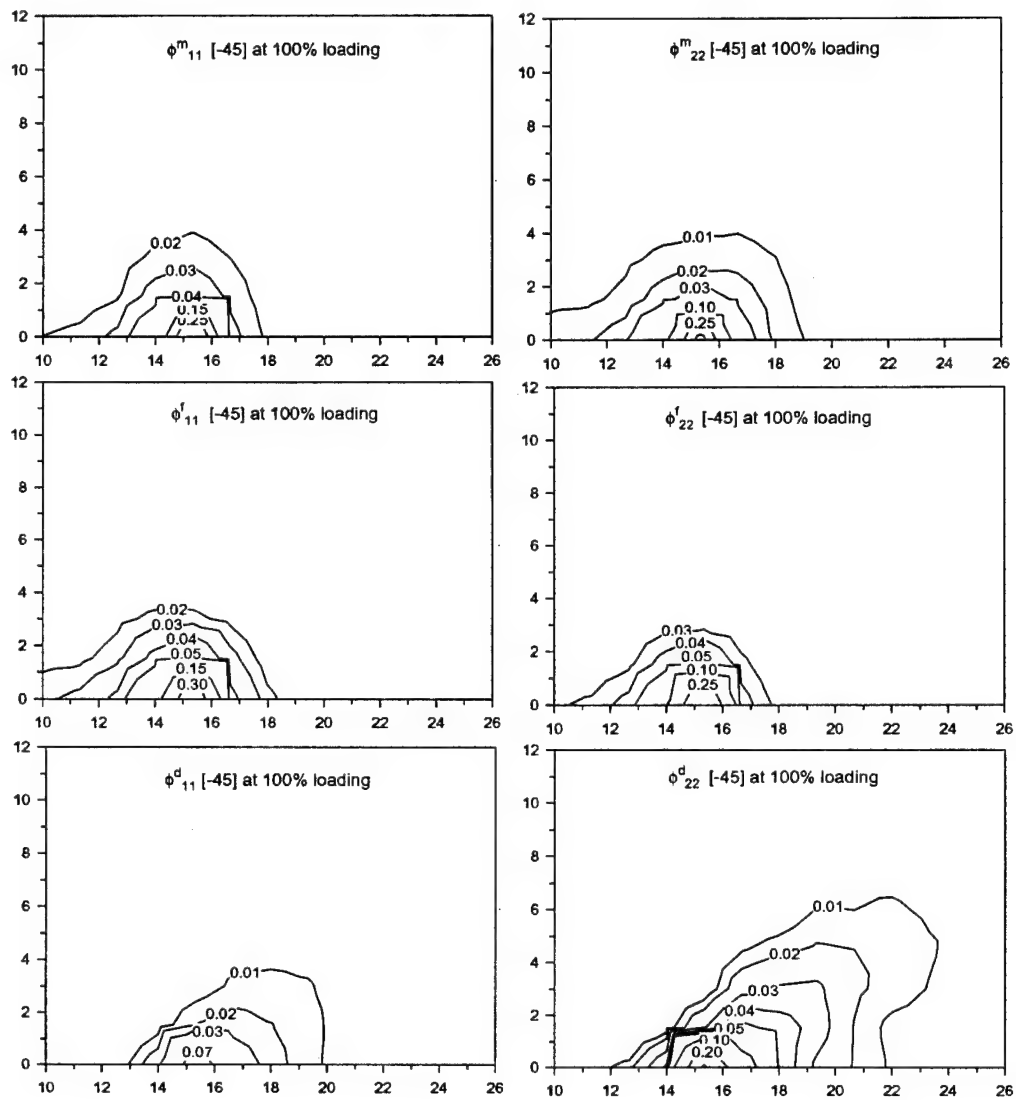
Figure 3.11: Damage contours around crack tip at the failure load for [−45] lamina.
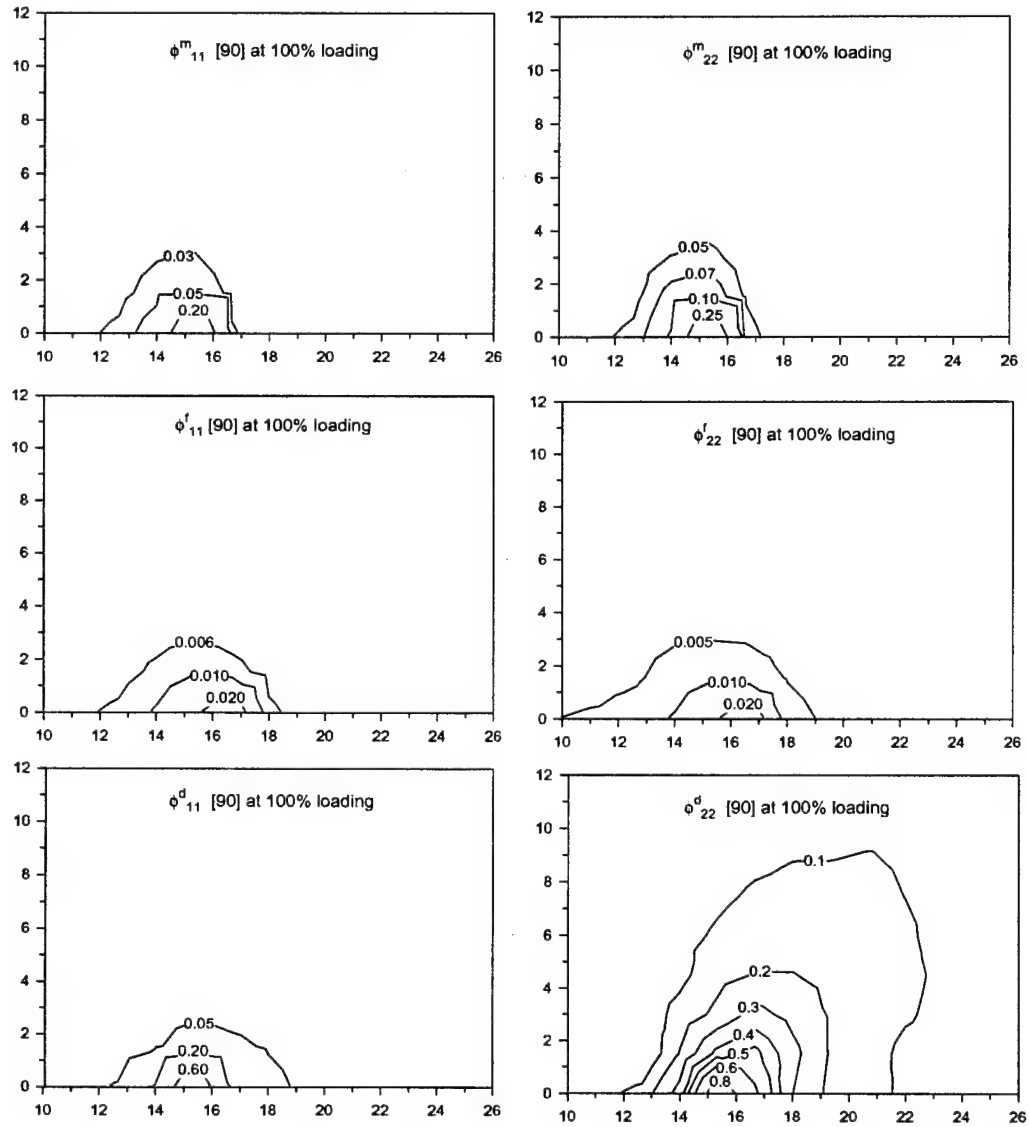
Figure 3.12: Damage contours around crack tip at the failure load for [0] lamina.
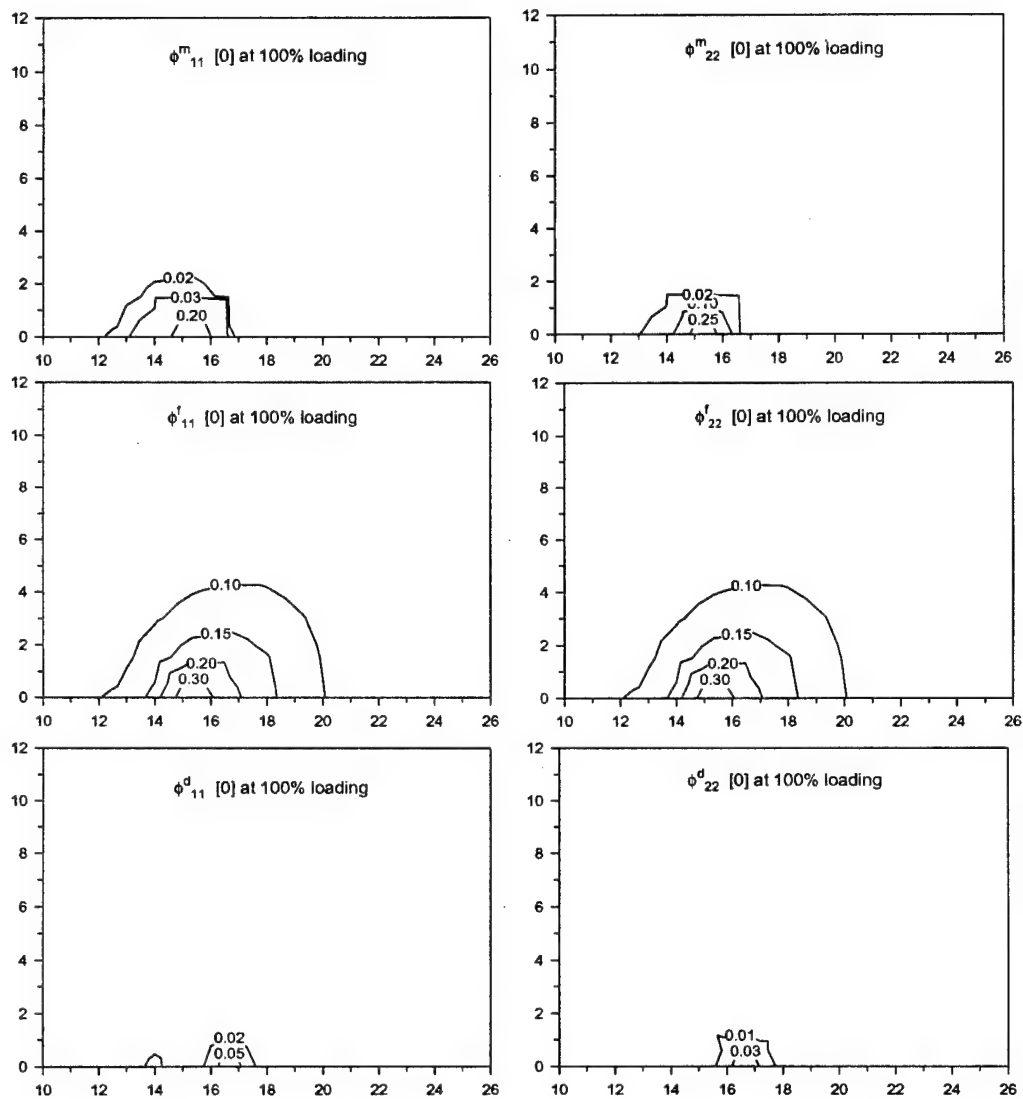
Figure 3.13: Damage contours around crack tip at the failure load for [90] lamina.

Chapter 4

A Damage Cyclic Plasticity Model For Metal Matrix Composites

4.1 Introduction

A mathematical model is presented here to simulate the behavior of metal matrix composites under cyclic proportional and non-proportional loading. This model incorporates both the phenomena of damage and cyclic plasticity. In this paper a brief description of the cyclic plasticity model is presented, [17], and based on this model the development of the damage based plasticity model is outlined.

The cyclic plasticity model is based on an anisotropic yield criterion proposed [17], [18]. The salient features of this criterion have been outlined along with some experimental comparisons. The model further uses a proposed non-associative flow rule and a modified form of the bounding surface model [19], for the case of anisotropic materials. This procedure involves the computation of the anisotropic plastic modulus. Experimental data from [20] and [21] have been used the for the computation of the various material parameters as well as comparison with experimental results.

All materials undergo damage, which is used synonymously with the degradation of the material's elastic stiffness here, as repeated loading takes place. To account for this phenomena a damage-plasticity model is presented here. This is based on the cyclic plasticity behavior blended with the damage model, [7] and [22].

In this paper, the development of the yield surface is presented at the outset followed by the cyclic plasticity model for the material treated as a continuum. Two different damage plasticity models are then outlined along with comparison of results in each case.

4.2 Description of the Yield Surface

An anisotropic yield surface of the form

$$\bar{M}_{ijkl}\bar{\sigma}_{ij}\bar{\sigma}_{kl} - 1 \;=\; 0 \tag{4.1}$$

is used here where $\bar{\sigma}_{ij}$ is the overall state of stress in the local coordinate axes. The local coordinate axes is defined as the principal axes of anisotropy of the material whereas the

global coordinate axes is the general axes along which loading is applied. Figure 4.1 shows the details of the local coordinate axes with respect to the global coordinate axes.
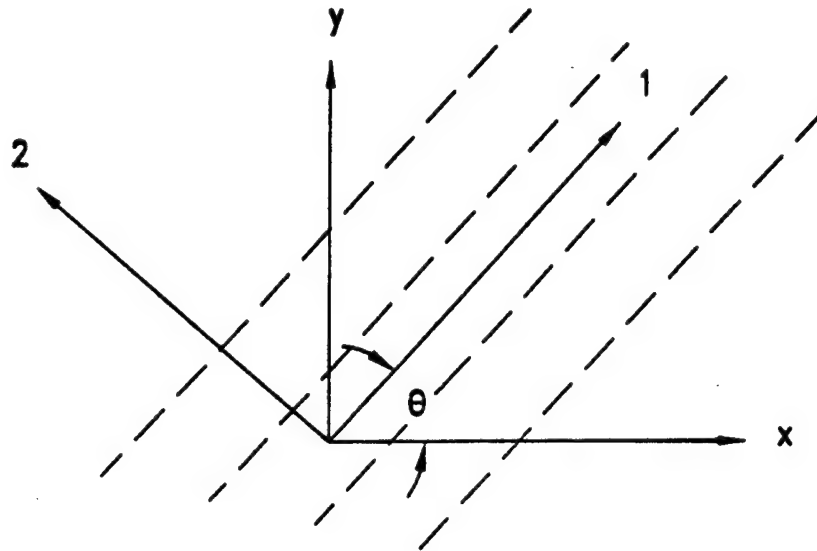
x-y: general axes of reference
1-2: principal material axes
— — direction of fiber

Figure 4.1: Local and General Axes of Reference for a Single Lamina

In order to accurately describe the yielding behavior of the orthotropic metal matrix composite a form for the fourth order anisotropic yield tensor $\bar{M}$ is shown here. This yield tensor has been dervied to satisfy certain criteria typical to metal matrix composites. It has been observed from experiments that the shear strength of anisotropic materials is independent of the axial yield strength of the material. Hence it is necessary to have three additional shear strength parameters in addition to the three principal axial strengths. Thus six strength parameters are used to describe the yielding behavior. It is also assumed at this stage that the axial strength in compression must be the same as that in tension.

It has also been observed that yielding in metal matrix composites is pressure dependent, [23]. Most of the commonly used forms of anisotropic yield criteria are pressure independent ones, [24], [25] .It is later shown that the yield criterion described here can be reduced to the pressure independent form by imposing suitable constraints to it.

The proposed form of the fourth order anisotropic yield tensor $\bar{M}$ based on the above conditions is as follows. It can be expressed as a function of two second order tensors a_{ij} and b_{ij} as,

$$\bar{M} = \bar{M}(a, b) \tag{4.2}$$

The functional form for $\bar{M}$ is defined as,

$$\bar{M}_{ijkl} = A(a_{ij}a_{kl}) + B(a_{ik}a_{jl}) + C(a_{il}a_{jk}) + D(b_{ij}b_{kl}) \tag{4.3}$$

72

where A, B, C and D are constants and a_{ij} and b_{ij} are functions of the 6 strength parameters k_i (i=1...6). Three of these parameters are directly related to the axial strengths (k_1, k_2, k_3) and the other three, k_4, k_5, k_6, are shear strength parameters used to define yielding for an anisotropic material. These parameters are measured and determined in the local coordinate axes. a_{ij} and b_{ij} are given as follows.

$$
a_{ij} = \begin{bmatrix} k_1 & 0 & 0 \\ 0 & k_2 & 0 \\ 0 & 0 & k_3 \end{bmatrix} \tag{4.4}
$$

$$
b_{ij} = \begin{bmatrix} 0 & k_4 & k_5 \\ k_4 & 0 & k_6 \\ k_5 & k_6 & 0 \end{bmatrix} \tag{4.5}
$$

By substituting equation (4.3) into equation (4.1), the yield function equation in component form in the local coordinate axes can be shown to be,

$$
\begin{aligned}
(A + B + C)(k_1^2 \bar{\sigma}_{11}^2 + k_2^2 \bar{\sigma}_{22}^2 + k_3^2 \bar{\sigma}_{33}^2) + \\
(2A)(k_1 k_2 \bar{\sigma}_{11} \bar{\sigma}_{22} + k_1 k_3 \bar{\sigma}_{11} \bar{\sigma}_{33} + k_2 k_3 \bar{\sigma}_{22} \bar{\sigma}_{33}) + \\
(2(B + C)k_1 k_2 + 4Dk_4^2)\bar{\sigma}_{12}^2 + (2(B + C)k_1 k_3 + 4Dk_5^2)\bar{\sigma}_{13}^2 + \\
(2(B + C)k_2 k_3 + 4Dk_6^2)\bar{\sigma}_{23}^2 - 1 = 0
\end{aligned} \tag{4.6}
$$

In the above equation the constants A, B, C, D are not material constants, but are chosen to suit various yield criteria as outlined below.

It can be shown that (4.6) reduces to the familiar von-Mises and Tresca yield criterion under the following combinations of the constants A,B,C and D.

1. For von-Mises (Isotropic) Criterion
   $A = -\frac{1}{9}$, $B = C = \frac{1}{6}$

2. For Tresca (Isotropic) Criterion
   $A = -\frac{1}{4}$, $B = C = \frac{1}{4}$

The values of these constants chosen here for this implementation are $A = -\frac{1}{9}, B = C = \frac{1}{6}$ and $D = \frac{1}{6}$. These values reduce the above equation to the following form.

$$
\begin{aligned}
F = &\; \frac{2}{9}(k_1^2 \bar{\sigma}_{11}^2 + k_2^2 \bar{\sigma}_{22}^2 + k_3^2 \bar{\sigma}_{33}^2) \\
&- \frac{2}{9}(k_1 k_2 \bar{\sigma}_{11} \bar{\sigma}_{22} + k_2 k_3 \bar{\sigma}_{22} \bar{\sigma}_{33} + k_1 k_3 \bar{\sigma}_{11} \bar{\sigma}_{33}) \\
&+ \frac{2}{3}(k_1 k_2 + k_4^2)\bar{\sigma}_{12}^2 + \frac{2}{3}(k_1 k_3 + k_5^2)\bar{\sigma}_{13}^2 + \frac{2}{3}(k_2 k_3 + k_6^2)\bar{\sigma}_{23}^2 - 1.0
\end{aligned} \tag{4.7}
$$

The convexity of the yield surface has been mathematically proven in [18], [17] with the only condition that the six parameters must be positive. Since these parameters also represent strength quantities physically, it is always positive.

Using the equation of stress transformation between the local and global axes of reference as follows,

$$\bar{\sigma}_{ij} = d_{ip}\sigma_{pq}d_{qj} \tag{4.8}$$

where d_{ij} are the coefficients of the orthogonal transformation matrix, the yield equation can be expressed in the global axes of reference as,

$$\sigma_{ij}M_{ijkl}\sigma_{kl} - 1 = 0 \tag{4.9}$$

Substituting for $\bar{\sigma}_{ij}$ in the yield equation (4.1) one obtains

$$\sigma_{pq}d_{ip}d_{jq}\bar{M}_{ijkl}d_{km}d_{ln}\sigma_{mn} - 1 = 0 \tag{4.10}$$

From the above equation M_{ijkl} can be derived to be,

$$M_{ijkl} = \bar{M}_{pqrs}d_{ip}d_{jq}d_{kr}d_{ls} \tag{4.11}$$

4.2.1 Comparison with Other Anisotropic Yield Surfaces

The described anisotropic yield surface is compared with two well known anisotropic criteria, that are frequently used for metal matrix composites, namely [26] and [27] criterion for transversely isotropic materials.

Hill's pressure-independent anisotropic yield criterion for orthotropic criterion is expressed as,

$$\begin{aligned} f = \; & F(\sigma_{yy} - \sigma_{zz})^2 + G(\sigma_{zz} - \sigma_{xx})^2 + H(\sigma_{xx} - \sigma_{yy})^2 + \\ & 2L\sigma_{yz}^2 + 2M\sigma_{xz}^2 + 2N\sigma_{xy}^2 - 1 \end{aligned} \tag{4.12}$$

The yield criterion described here is a pressure dependent yield criterion. In order reduce it to a pressure independent one, deviatoric stresses along with condition that the fourth order anisotropic tensor $\boldsymbol{M}$ must satisfy the condition,

$$M_{iikk} = 0 \tag{4.13}$$

By applying this condition to the proposed tensor $\boldsymbol{M}$ we arrive at the condition,

$$k_1^2 + k_2^2 + k_3^2 - k_1k_2 - k_1k_3 - k_2k_3 = 0 \tag{4.14}$$

By taking the deviatoric stresses and on expanding and comparing the two yield criteria it can be shown that,

$$\begin{aligned} H + G &= \frac{2}{27}k_1^2 \\ H + F &= \frac{2}{27}k_2^2 \\ F + G &= \frac{2}{27}k_3^2 \\ 2L &= \frac{2}{3}(k_1k_2 + k_4^2) \\ 2M &= \frac{2}{3}(k_1k_3 + k_5^2) \\ 2N &= \frac{2}{3}(k_2k_3 + k_6^2) \end{aligned} \tag{4.15}$$

The first three terms in the above equation represent the uniaxial yield strength along the three axes of anisotropy while the last three terms represent the corresponding shear strengths in both criteria.

The correspondence between the Mulhern, Rogers and Spencers parameters and criterion and the criterion described here has been shown in detail in [17, 18].

4.2.2 Numerical Simulation of the Anisotropic Yield Surface

A numerical simulation is done to evaluate the values of the parameters of the proposed yield surface, from the experimental data obtained from boron-aluminum composite tubular specimen having unidirectional lamina, [20] and [21]. The fibers in the laminae of the tube are aligned parallel to the axis of the tube. The specimen is subjected to different loading patterns by applying axial force, torque and internal pressure in order to determine the yield surfaces in the $(\sigma_{11} - \sigma_{21})$ and $(\sigma_{22} - \sigma_{21})$ stress planes, where σ_{11} is the stress along the fiber direction, σ_{22} is the normal stress transverse to the fiber direction and σ_{21} is the longitudinal shear stress. The parameters which have been evaluated from the experimental data are then used to generate the corresponding yield surfaces, which are then compared with those obtained from experiments.

The orientation of the fibers along the axis of the tube is represented mathematically be $\eta = (1,0,0)$. The yield surface equation (4.1) is then reduced to component form for transversely isotropic material case where $k_2 = k_3$ to get,

$$F = \frac{2}{9}k_1^2\sigma_{11}^2 + \frac{2}{9}k_2^2(\sigma_{22}^2 + \sigma_{33}^2) - \frac{2}{9}k_1k_2\sigma_{11}(\sigma_{22} + \sigma_{33})$$
$$-\frac{2}{9}k_2^2\sigma_{22}\sigma_{33} + \frac{2}{3}(k_1k_2 + k_4^2)(\sigma_{12}^2 + \sigma_{13}^2) + \frac{2}{3}(k_2^2 + k_6^2)\sigma_{23}^2 - 1 \qquad (4.16)$$

If only σ_{11} and σ_{21} are the non-zero stresses, the above equation can be reduced to,

$$F = \frac{2}{9}k_1^2\sigma_{11}^2 + \frac{2}{3}(k_1k_2 + k_4^2)\sigma_{21}^2 - 1 \qquad (4.17)$$

A similar equation can be written in the $(\sigma_{22} - \sigma_{21})$ space also. The parameter k_1 is determined from the yield stress along the σ_{11} axis and k_2 from that along the σ_{22} axis. From the third yield stress namely along the σ_{21} axis we can then determine k_4 using the above equation. From the experimental data, the values of initial yield stress have been measured as, $\sigma_{11}^Y = 87.90\ MPa$, $\sigma_{22}^Y = 44.70\ MPa$ and $\sigma_{21}^Y = 17.90\ MPa$.

Using the above data the values for k_1, k_2 and k_4 are evaluated to be,

- $k_1 = \frac{1}{41.47}$

- $k_2 = \frac{1}{21.09}$

- $k_4 = \frac{1}{16.81}$

The yield surfaces that is represented by this model is generated using the above parameter values. Figure 4.2 shows the model generated surfaces along with the points obtained from experimental data, for initial yield surfaces in this stress space.

Figure 4.2 shows why the necessity to have additional parameters to represent the non-conformal effect of shear strength. The figure shows the curves corresponding to a model having only the parameters k_1 and k_2. It is observed that although the yielding along the axial directions are correctly simulated, the shear strength is overestimated. The introduction of the shear strength parameter k_4 corrects this deficiency and allows for the correct representation of the observed phenomena. The parameter values for subsequent yield surfaces have also been computed from the experimental data available. Figure 4.3 shows the model simulated yield surfaces as compared to the experimental subsequent yield surfaces.
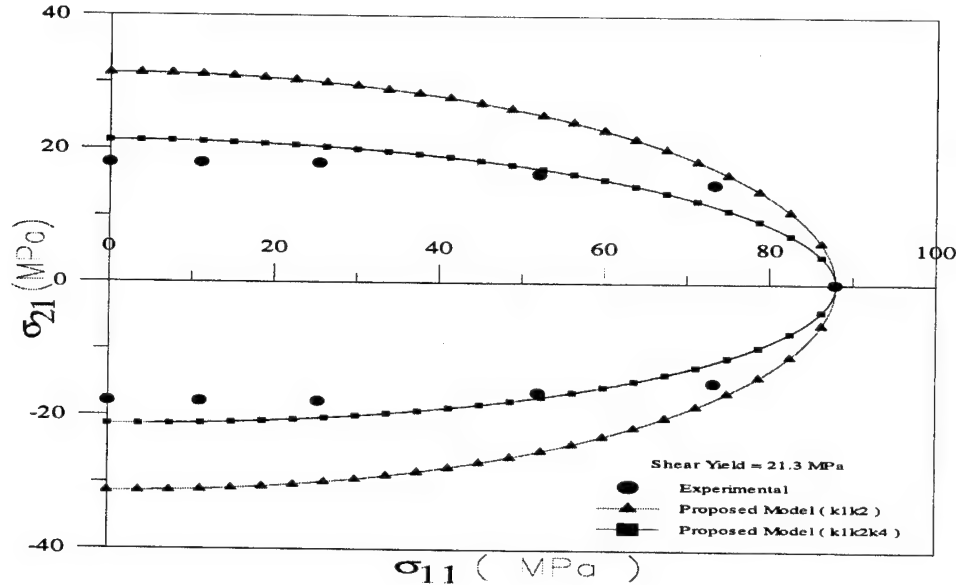
Figure 4.2: Comparison of Initial Yield Surface in $\sigma_{11} - \sigma_{12}$ space

4.3 Continuum Cyclic Plasticity Model

Using the anisotropic yield surface outlined above, a cyclic plasticity model is described here, for the metal matrix composite treated as a continuum. The plasticity model uses a kinematic hardening rule along with a non-associative flow rule. These along with the constitutive equations are described below.

4.3.1 Elastic Behavior

The elastic behavior of the composite material, treated as a homogeneous continuum with transversely isotropic properties has been defined in [28] and is used here. The linear constitutive relation is expressed as

$$\sigma_{ij} = C_{ijkl}\epsilon_{kl} \tag{4.18}$$

where C is the fourth order elastic stiffness tensor relating the symmetric second order tensors σ and ϵ of stress and strain respectively. For a trasnversely isotropic material the
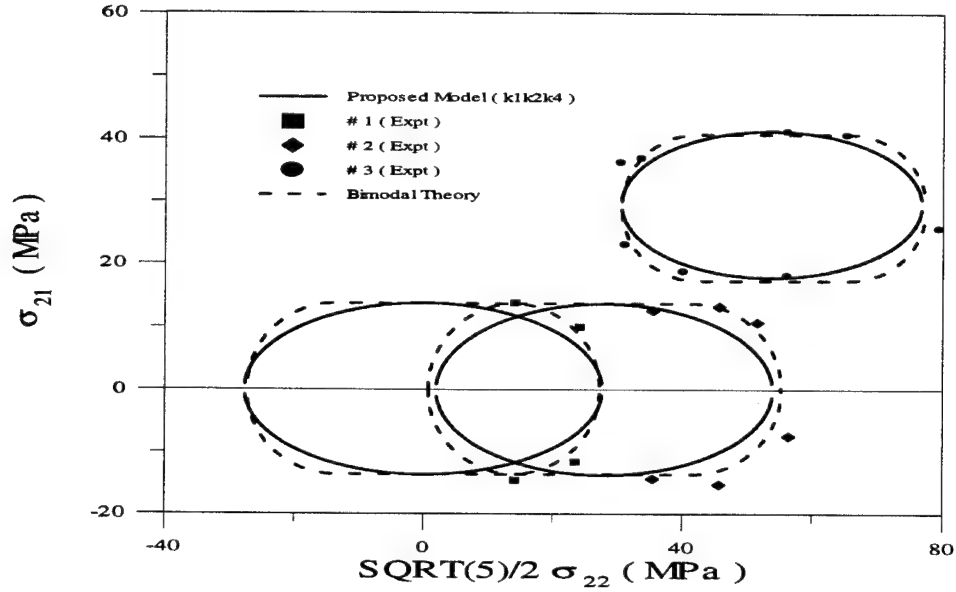
Figure 4.3: Comparison of Subsequent Yield Surfaces in $\sigma_{22} - \sigma_{12}$ space

fourth order elastic stiffness tensor is given as follows.

$$C_{ijkl} = Kt_{ij}t_{kl} + El_{ij}l_{kl} + 2mE^3_{ijkl} + 2pE^4_{ijkl} \tag{4.19}$$

where,

$$t_{ij} = m_{ij} + 2\nu l_{ij} \tag{4.20}$$

$$l_{ij} = \eta_i\eta_j \tag{4.21}$$

$$m_{ij} = \delta_{ij} - \eta_i\eta_j \tag{4.22}$$

$$E^3_{ijkl} = \frac{1}{2}\left[m_{ik}m_{jl} + m_{jk}m_{il} - m_{ij}m_{kl}\right] \tag{4.23}$$

$$E^4_{ijkl} = \frac{1}{2}\left[m_{ik}l_{jl} + m_{il}l_{jk} + m_{jl}l_{ik} + m_{jk}m_{il}\right] \tag{4.24}$$

and K is the plane-strain bulk modulus, m is the transverse shear modulus, p is the axial shear modulus and E and ν are the Young's Modulus and Poisson's ratio respectively, when the material is loaded in the fiber direction. For a transversely isotropic material the plane-strain bulk modulus can be defined in terms of the other four elastic constants.

4.3.2   Kinematic Hardening

Kinematic hardening is accounted for by modifying the form of the yield surface as follows.

$$f = (\sigma_{ij} - \alpha_{ij})M_{ijkl}(\sigma_{kl} - \alpha_{kl}) - 1.0 \tag{4.25}$$

The evolution equation for the backstress is based on the Phillips rule and can be expressed as follows.

$$\dot{\alpha}_{ij} = \|\dot{\alpha}_{rs}\|l_{ij} \tag{4.26}$$

where l_{ij} is a unit tensor along the stress rate direction.

77

4.3.3 Non Associative Flow Rule

It is observed that the determination of plastic strains, for any anisotropic material in general, and an MMC in particular must adopt a non-associative flow rule. This has been demonstrated experimentally [20], [21]. They have also observed that the direction of plastic strains tend to be more inclined towards the shear direction in a combined transverse tension-shear loading situation. Also plastic inextensibility along the fiber direction is an accepted MMC behavior.

A plastic potential function is defined here, the form of which is based on the proposed yield function. To determine the plastic strain increments ($\acute{\epsilon}_{ij}''$) a non- associative flow rule is used as follows,

$$\acute{\epsilon}_{ij}'' = \dot{\Lambda}\frac{\partial G}{\partial \sigma_{ij}} \tag{4.27}$$

where G is the plastic potential function. The potential function is defined here as a function of the yield function g and the constrained yield function as follows,

$$G = \omega f + (1.0 - \omega)g, \quad 0 \leq \omega \leq 1.0 \tag{4.28}$$

The constrained yield function g is defined such that it satisfies the condition of plastic inextensibility along the direction of fiber defined by $\boldsymbol{\eta}$. The function g is defined using the fourth order anisotropic yield tensor $\bar{M}$ and a constrained stress term r_{ij} such that,

$$g = r_{ij}\bar{M}_{ijkl}r_{kl} - 1 \tag{4.29}$$

The constraint that is introduced in the stress term is that the plastic strain increment is independent of the component of stress along a specified direction (defined here by η_i) . Following the procedure outlined in [29] the constraint is incorporated into the stress term as follows,

$$r_{ij} = \sigma_{ij} - T\eta_i\eta_j \tag{4.30}$$

where $T\eta_i\eta_j$ is the reaction to an inextensibility constraint along the direction $\boldsymbol{\eta}$.

Based on the flow rule, one can define a second order tensor representing the direction of plastic strains as follows.

$$n_{ij} = \omega n_{ij}^f + (1.0 - \omega)n_{ij}^g \tag{4.31}$$

where ω is a non-associativity parameter that can be determined from experiments as explained in [17] and n_{ij}^f and n_{ij}^g are unit normals to the yield and potential surfaces respectively.

4.3.4 Evaluation of Elasto-Plastic Stiffness Matrix

The fourth order elasto-plastic stiffness matrix for the MMC treated as a continuum is determined by incorporating all the procedures outlined above. Small deformations as well as rate independency of plastic strains is assumed. This allows one to use an additive decomposition of the incremental strain tensor $d\epsilon_{ij}$ such that,

$$d\epsilon_{ij} = d\epsilon_{ij}' + d\epsilon_{ij}'' \tag{4.32}$$

78

where $d\epsilon'_{ij}$ is the elastic part and $d\epsilon''_{ij}$ is the plastic part of the strain tensor. The incremental stress-strain relations can be expressed as follows.

$$d\sigma_{ij} = C_{ijkl}d\epsilon'_{kl} \tag{4.33}$$

$$= C_{ijkl}(d\epsilon_{kl} - d\epsilon''_{kl}) \tag{4.34}$$

Using equation (4.27) for the plastic strain part we can write the above equation as follows.

$$\sigma_{ij} = C_{ijkl}(d\epsilon_{kl} - \frac{d\sigma}{H}n_{kl}) \tag{4.35}$$

Equation (4.35) may also be expressed as follows.

$$\dot{\sigma}_{ij} = C_{ijkl}\dot{\epsilon}_{kl} - C_{ijkl}n_{kl}\dot{\sigma}/H \tag{4.36}$$

Taking the inner product of both sides with n_{ij} one obtains,

$$\dot{\sigma}_{ij}n_{ij} = C_{ijkl}n_{ij}\dot{\epsilon}_{kl} - C_{ijkl}n_{ij}n_{kl}\dot{\sigma}/H \tag{4.37}$$

$$= \dot{\sigma} \tag{4.38}$$

Or,

$$\dot{\sigma}\left(1 + \frac{C_{ijkl}n_{ij}n_{kl}}{H}\right) = C_{ijkl}n_{ij}\dot{\epsilon}_{kl} \tag{4.39}$$

From the above equation one can express $\dot{\sigma}$ as follows.

$$\dot{\sigma} = \frac{C_{ijkl}n_{ij}\dot{\epsilon}_{kl}}{H + C_{ijkl}n_{ij}n_{kl}}H \tag{4.40}$$

Hence the expression for plastic strains using equation (34) can be written explicitly as follows.

$$\epsilon''_{ij} = \frac{C_{abcd}n_{ab}d\epsilon_{cd}}{H + C_{pqrs}n_{pq}n_{rs}}n_{ij} \tag{4.41}$$

Substituting this in the equation for the incremental stress-strain relations one obtains,

$$d\sigma_{ij} = C_{ijkl}\left[d\epsilon_{kl} - \frac{C_{abcd}n_{ab}d\epsilon_{cd}}{H + C_{pqrs}n_{pq}n_{rs}}n_{kl}\right] \tag{4.42}$$

$$= C_{ijkl}d\epsilon_{kl} - \frac{C_{ijkl}C_{abcd}n_{ab}n_{kl}d\epsilon_{cd}}{H + C_{pqrs}n_{pq}n_{rs}} \tag{4.43}$$

Interchanging the indices kl with cd in the second term of the above equation one obtains,

$$d\sigma_{ij} = C_{ijkl}d\epsilon_{kl} - \frac{C_{ijcd}C_{abkl}n_{ab}n_{cd}d\epsilon_{kl}}{H + C_{pqrs}n_{pq}n_{rs}} \tag{4.44}$$

or,

$$d\sigma_{ij} = D^{EP}_{ijkl}d\epsilon_{kl} \tag{4.45}$$

where D^{EP}_{ijkl} is the elasto-plastic stiffness of the material and is expressed as follows.

$$D^{EP}_{ijkl} = C_{ijkl} - \frac{C_{ijcd}n_{cd}C_{abkl}n_{ab}}{H + C_{pqrs}n_{pq}n_{rs}} \tag{4.46}$$

79

4.3.5 Anisotropic Plastic Modulus

For initially anisotropic or orthotropic materials the asymptotic value of the plastic modulus H need not and in most cases will not be the same for all points on the yield surface. For materials where we assume the behavior in tension and compression to be similar, it is reasonable to assume that at mirror image points of the yield surface , this asymptotic value of the plastic modulus is the same.
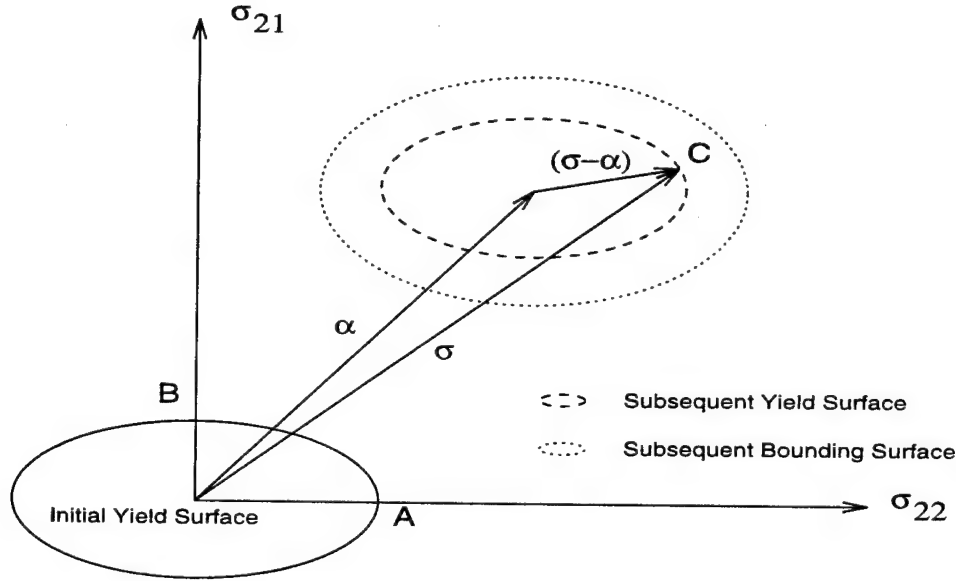
Figure 4.4: Illustration to Explain Plastic Modulus Determination for an Anisotropic Material

In Figure 4.4, which shows the yield and the bounding surfaces in the $\sigma_{22} - \sigma_{12}$ stress space, points **A** and **B** are the location of the stress points for initial yielding for loading in the σ_{22} and σ_{12} respectively. From the two uniaxial stress-strain behaviors, it is observed that the plastic modulus and hence the values of the three parameters that are required to determine the plastic modulus, are different for loading paths along the two directions.

A modified form of the bounding surface model is used to model the behavior of the composite material under cyclic loading situations. The main idea here is the determination of the plastic modulus H. The observed values of the parameters involved in the determination of the plastic modulus are different along different loading directions. This could be modelled by using tensors in the form of second order tensors such as δ_{ij}^{in}, h_{ij} and H_{ij}^*. These are then converted to a scalar valued form by taking the inner product of these tensors with another second order tensor ρ_{ij} and representing the result as follows.

$$\tilde{H}^* = H_{ij}^* \rho_{ij} \tag{4.47}$$

$$\tilde{h} = h_{ij} \rho_{ij} \tag{4.48}$$

$$\tilde{\delta}^{in} = \delta_{ij}^{in} \rho_{ij} \tag{4.49}$$

The expression for the plastic modulus can then be expressed as

$$H = \tilde{H}^* + \tilde{h}\delta/(\tilde{\delta}^{in} - \delta) \tag{4.50}$$

80

where δ is the distance between the stress point on the yield surface and the image point on the bounding surface. In this model ρ_{ij} is chosen to be the same direction as l_{ij}. Further details of this selection is presented in [30].

The initial bounding surface used here is an identical expansion of the initial yield surface. The bounding surface is expressed as

$$f^b(\sigma_{ij}^b, \beta_{ij}, \bar{a}_{ij}, \bar{b}_{ij}) \;=\; 0 \tag{4.51}$$

where σ_{ij}^b is the image point on the bounding surface and β_{ij} is its center. The evolution of the center of the bounding surface in the stress space as loading continues is related to the evolution of backstress for the yield surface as well the relative distance between the stress point and the image point.

4.3.6   Experimental Comparison and Discussions

From the data presented in the paper in [21], the following values for the bounding surface have been evaluated. $\sigma_{11}^b = 196.0$ MPa, $\sigma_{22}^b = 91.5$ MPa and $\sigma_{21}^b = 34.0$ MPa. This results in the computed values of the initial bounding surface parameters as $\bar{k}_1 = 0.0108$, $\bar{k}_2 = 0.0232$ and $\bar{k}_4 = 0.0323$. The values for k_3, k_5 and k_6 are not needed here and have been taken to be zero.

The numerical evaluation of the plastic modulus constants is the next step in this process, i.e. the evaluation of H_{ij}^* and h_{ij}. Different values of these constants are evaluated from experimental results of the uniaxial stress-plastic strain curves along different stress directions. The values of these constants have been evaluated as follows. $H_{11}^* = 1,600,000$MPa , $H_{22}^* = 12000$MPa and $H_{12}^* = 6000$MPa and the values of the other parameter h_{ij} are $h_{11} = 9,650,000$, $h_{22} = 90,000$ and $h_{21} = 40,000$MPa respectively. The other values of this tensor are assumed to be zero.

In order to incorporate the non-associativity of the flow rule that has been built into the model, the value of ω has been chosen, by trial and error, as $\omega = 0.5$.

Figure 4.5 shows the comparison between between experimentally obtained and model generated $\sigma_{22} - \epsilon_{22}''$ curve and Figure 4.6 shows the same of $\sigma_{21} - 2\epsilon_{12}''$. From the comparison of the experimental results in [20], [21] and the model generated stress-strain curves, a reasonably good correlation is observed.

The tendency for ratchetting to occur for cyclic loading, for 5 cycles of loading path has also been observed. But the tendency to stabilize have been different for the experimental and model predicted results. This is because a drastic degradation of elastic modulus has been observed in the experimental results.

A significant feature of this model is the usage of a non-associative flow rule. In order to demonstrate its significance, the model is run with the same loading situation, but with $\omega = 1.0$, which results in the usage of an associated flow rule. Figures 4.7 & 4.8 show the comparison of model and experimental results for this case. For a pure associative flow rule ($\omega = 1.0$) it is seen that plastic strains ϵ_{22}'' have been overpredicted. A factor of $\omega = 0.5$ which incorporated non-associativity into the model has been successfully used to predict the plastic strains reasonably in this direction.

Results are also shown for one simulated loading situation. Figure 4.9 shows the results for radial cyclic loading in the $\sigma_{22} - \sigma_{21}$ stress space. From these results it is observed that

81
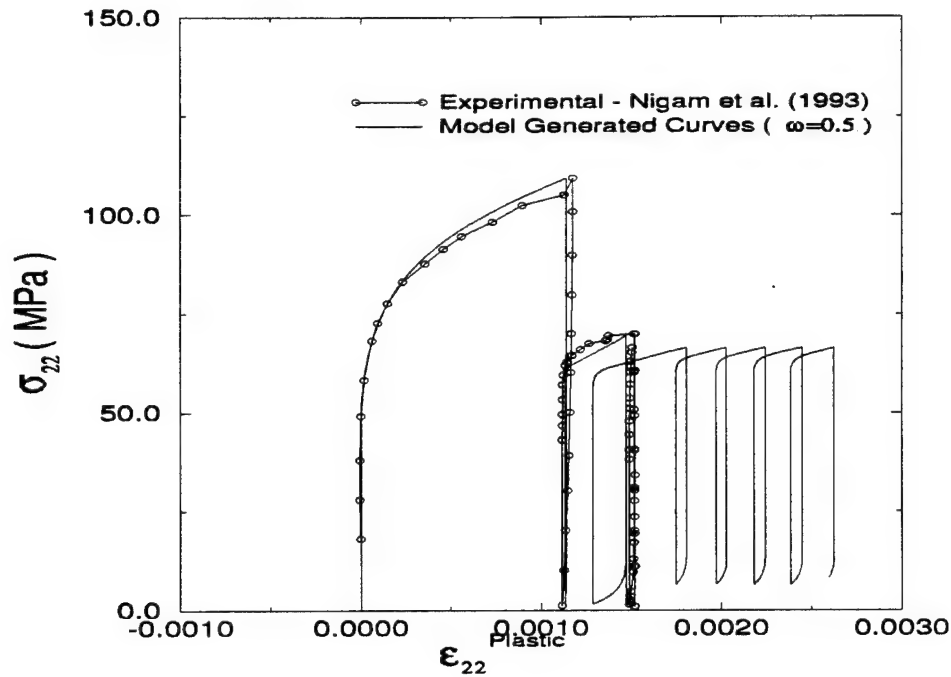
Figure 4.5: Transverse Stress-Plastic Strain Comparison for Non-Associative Flow Rule

the model is able to predict different behavior in different stress spaces during the loading process.

4.3.7 Comparison with Other Existing Models

The plastic strains predicted by the model presented here has been compared with those predicted by two micromechanical models, namely the Periodic Hexagonal Array (PHA) model, [31], and the self-consistent scheme,[26] and [32] using the Mori-Tanaka averaging scheme for the evaluation of the concentration factors, [33]. The data for the self-consistent and the PHA model have been taken from the paper by Lagoudas, [33].

Figure 4.6 shows the comparison of the shear stress-plastic shear strain curves generated by the above two mentioned models and the presented model along with those from the experimental data. It is seen that while the Mori-Tanaka and the PHA model results underpredict the plastic strains, the presented model using the non-associative flow rule comes closer in its prediction.

4.4 Damage

4.4.1 Description of Proposed Damage Models

In this work the metal matrix composite is assumed to consist of an elasto-plastic matrix with continuous aligned uni-directional elastic fibers. The composite system is restricted to small deformations with small strains. Two different approaches to model the damage
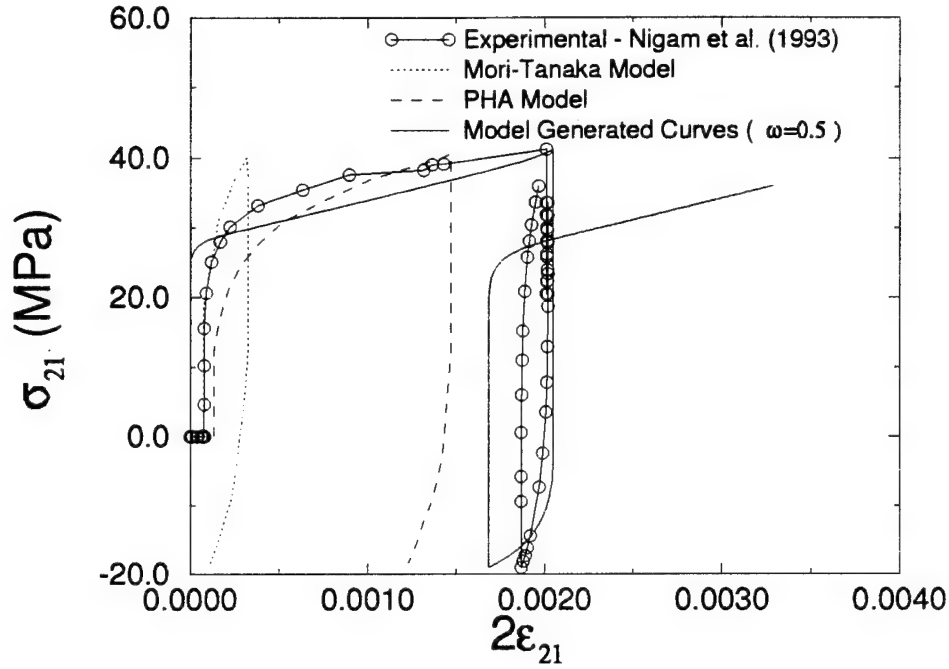
Figure 4.6: Shear Stress-Plastic Strain Comparison for Non-associative Flow Rule

behavior are presented here in this work.

In both these approaches the effective configuration is defined as a fictitious state with all damage removed, and the damaged configuration is the actual state of the material.

In the first approach the MMC is modeled using a 'Continuum Damage' model, wherein the MMC is treated as a continuum . The elasto-plastic behavior of the continuum is modeled using the cyclic plasticity model described earlier, applied to the effective continuum material and the damage transformation of this fictitious undamaged continuum to the damaged configuration is then obtained using the damage model. The damaged configuration is termed as C whereas the fictitious undamaged configuration is termed as $\bar{C}$.

In the second approach the MMC is treated as a micromechanical combination of an 'in-situ' plastic matrix and an elastic fiber. It is assumed that the in-situ behavior of the matrix material in the presence of the dense fibers is different from what it would be in the absence of fibers. Here only the in situ plasticity behavior of the matrix is characterized by the continuum cyclic-plasticity-composite model shown earlier. The initial undamaged and undeformed configuration of the composite material is denoted by C_o, and the damaged and deformed configuration after the body is subjected to a set of external agencies, is denoted by C. The fictitious configuration, $\bar{C}$, of the composite system is obtained from C by removing all the damage. $\bar{C}$ is termed as the effective configuration which is based on the effective stress concept, [1]. The sub configurations of $\bar{C}$ of the matrix and fibers are denoted by $\bar{C}^m$ and $\bar{C}^f$ respectively. Figure 4.10 shows the steps involved in this development.

The equations of continuum damage mechanics are then applied to the overall undamaged configuration $\bar{C}$ in order to obtain the effective damaged quantities in the overall configuration C.
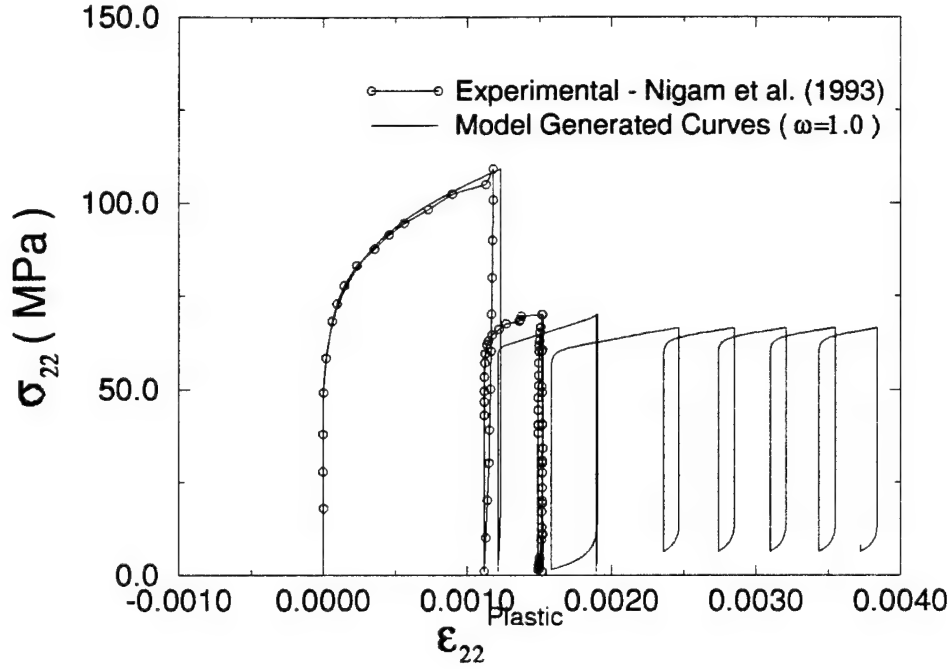
Figure 4.7: Shear Stress-Plastic Strain Comparison for Associative Flow Rule

The primary constitutive relationship in the effective configuration in incremental form can be expressed as,

$$\dot{\bar{\sigma}}_{ij} = \bar{D}_{ijkl} : \dot{\bar{\epsilon}}_{kl} \tag{4.52}$$

4.4.2 Damage Effect Tensor

The damage of the material is quantified through the fourth-order damage effect tensor M. This tensor reflects all kinds of damage such as matrix cracking and fiber breakage damage between the matrix and the fiber. This overall damage effect tensor M can be related to the local damage effect tensors such as

$$M = (\bar{c}^m M^m : B^m + \bar{c}^f M^f : B^f) \tag{4.53}$$

where M^m and M^f are the respective local damage effect tensors reflecting matrix damage and fiber damage [7]. A linear transformation is assumed between the Cauchy stress tensors such that

$$\bar{\sigma} = M : \sigma \tag{4.54}$$

[3] has shown that M can be represented by a 6x6 matrix as a function of a symmetric second order tensor ϕ such that

$$[M] = [M(I_2 - \phi)] \tag{4.55}$$

where I_2 is the second-rank identity tensor. The effective Cauchy stress need not be symmetric or frame invariant under the given transformation. However, once the effective Cauchy stress is symmetrized, it can be shown that it satisfies the frame invariance principle [22].

84
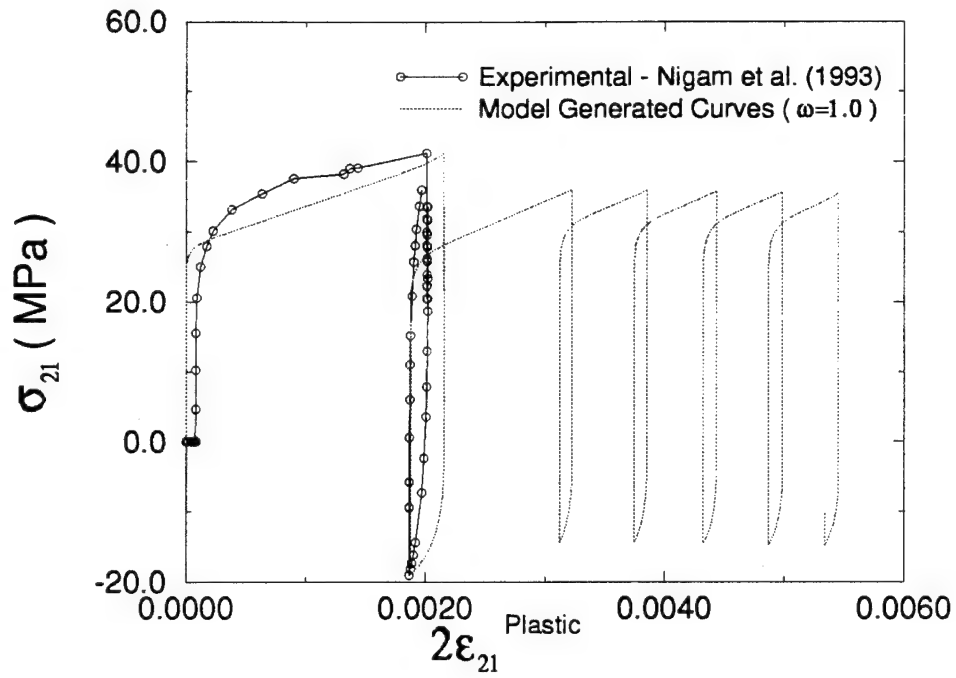
Figure 4.8: Transverse Stress-Plastic Strain Comparison for Associative Flow Rule

4.4.3 Anisotropic Damage Criterion

The damage criterion g is given in terms of the tensorial damage hardening parameter h and the generalized thermodynamic force Y conjugate to the damage tensor ϕ and a term γ which is defined in the thermodynamic force space such that

$$g \equiv (Y_{ij} - \gamma_{ij}) : P_{ijkl} : (Y_{kl} - \gamma_{kl}) - 1 = 0 \tag{4.56}$$

The fourth order tensor P is expressed in terms of the second order tensor h such that

$$P_{ijkl} = h_{ij}^{-1} h_{kl}^{-1} \tag{4.57}$$

A new and simplified form of the tensor h is given in terms of the second order tensor u, V and ϕ as follows

$$h_{ij} = (u_{ij} + V_{ij}) \tag{4.58}$$

The tensors u and V are scalar forms of isotropic materials originally proposed in [34] The tensors are given by

$$u = \begin{bmatrix} \lambda_1 q \left(\frac{\kappa}{\lambda_1}\right)^r & 0 & 0 \\ 0 & \lambda_2 q \left(\frac{\kappa}{\lambda_2}\right)^r & 0 \\ 0 & 0 & \lambda_3 q \left(\frac{\kappa}{\lambda_3}\right)^r \end{bmatrix} \tag{4.59}$$

85
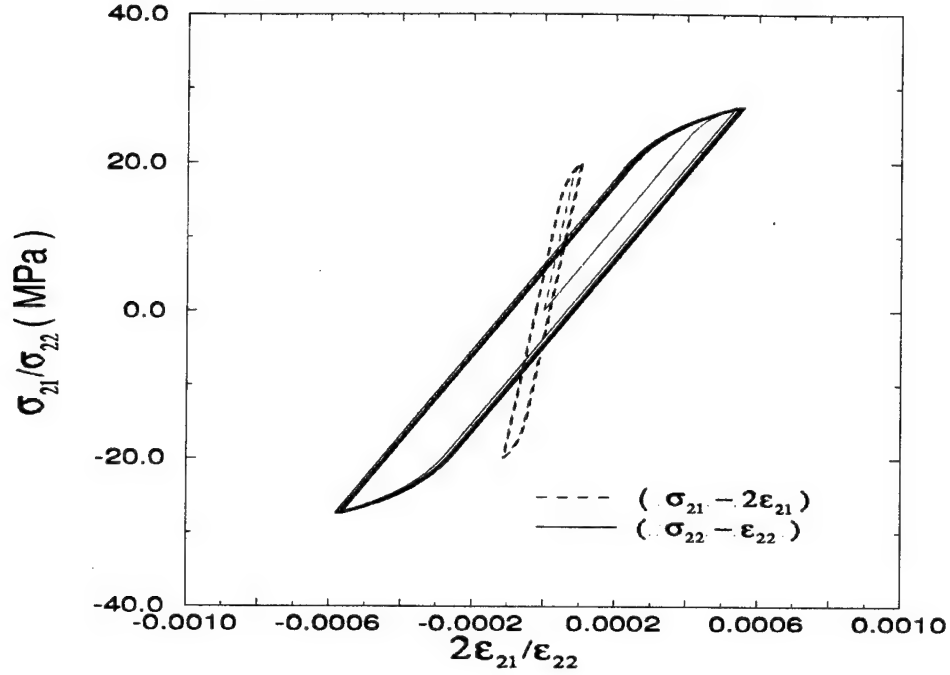
Figure 4.9: Stress-plastic strain - Simulated Radial stress loading in $\sigma_{22} - \sigma_{21}$

and

$$V = \begin{bmatrix} \lambda_1 v_1^2 & 0 & 0 \\ 0 & \lambda_2 v_2^2 & 0 \\ 0 & 0 & \lambda_3 v_3^2 \end{bmatrix} \tag{4.60}$$

The material parameters λ_1, λ_2 and λ_3 are Lame's constants for anisotropic materials and are related to the elasticity tensor E for an orthotropic material expressed by the 6×6 matrix, [7]. The material parameters v_1, v_2 and v_3 define the initial threshold against damage for the orthotropic material. These are obtained from the constraint that the onset of damage corresponds to the stress level at which virgin material starts exhibiting nonlinearity. The scalar damage hardening parameter κ is given by

$$\kappa = \int_0^t -Y : \dot{\phi} dt \tag{4.61}$$

Finally the material parameters r and q are obtained by comparing theory with experimental results.

4.4.4 Evolution of γ and $\dot{\phi}$

A new term γ has been introduced here in the definition of the damage criterion g in equation (4.56). This term is analogous to the backstress term in the stress-space yield criterion. It represents the translation of the damage surface as loading progress akin to kinematic hardening.

86
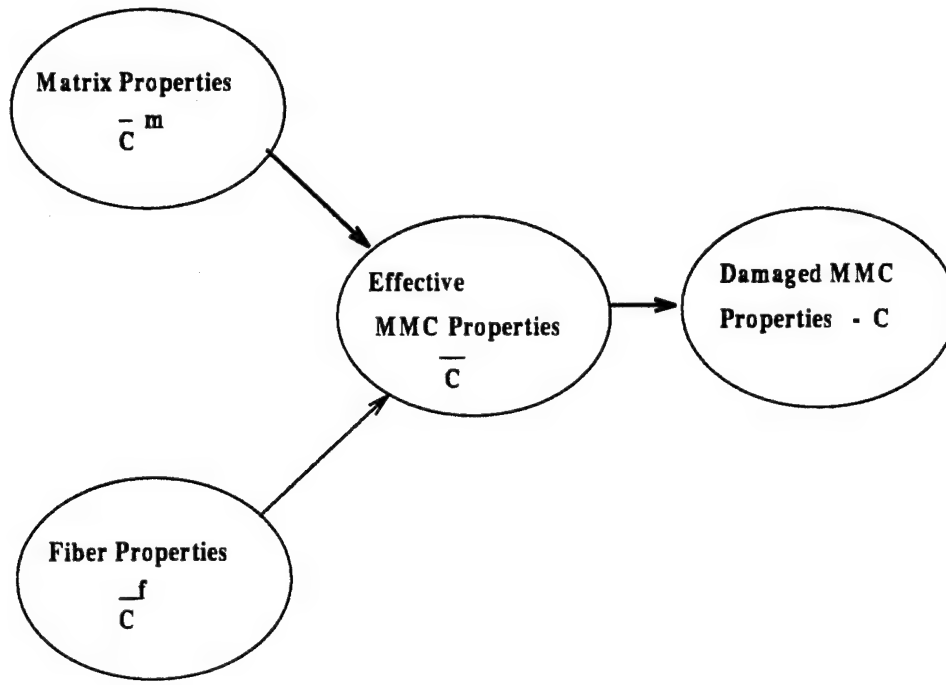
Figure 4.10: Schematic Diagram Depicting the Micromechanical Model

The evolution of the term γ in the anisotropic damage criterion equation is needed in order to account for the motion of the damage surface in the Y space. This is dependent on the evolution of damage itself. Hence it can be expressed mathematically as follows,

$$\dot{\gamma} = c\dot{\phi} \tag{4.62}$$

Since Y is negative γ too has to be negative. It has been found that it is suitable to adopt a value of -1 for the value of c. The negative sign is adopted because Y itself is a negative quantity as defined in equation (4.65)

The evolution of the damage variable ϕ is defined as follows:

$$\dot{\phi}_{ij} = \dot{\Lambda}_2 \frac{\partial g}{\partial Y_{ij}} \tag{4.63}$$

where g is the function representing the damage criterion.

The generalized thermodynamic free energy $\mathbf{Y}$ is assumed to be a function of the elastic-component of the strain tensor ϵ' and the damage tensor ϕ, or the stress σ and ϕ

$$Y = Y(\epsilon', \phi) \quad or \quad Y = Y(\sigma, \phi) \tag{4.64}$$

Making use of the evolution equations for $\mathbf{Y}$

$$\dot{Y}_{ij} = \frac{\partial Y_{ij}}{\partial \sigma_{mn}} \dot{\sigma}_{mn} + \frac{\partial Y_{ij}}{\partial \phi_{kl}} \dot{\phi}_{kl} \tag{4.65}$$

Making use of the energy equivalence principle, one obtains a relation between the damaged elasticity tensor $\mathbf{E}$ and the effective undamaged elasticity tensor $\bar{\mathbf{E}}$ such that [22]

$$E^{-1}_{mnkl}(\phi) = M_{uvmn}(\phi)\bar{E}^{-1}_{uvpq}M_{pqkl}(\phi) \tag{4.66}$$

87

4.5 Damage-Plasticity Constitutive Model

The stiffness tensor $\mathbf{D}$ for the damaged material now derived for isothermal conditions and in the absence of rate dependent effects. Making use of the incremental form of equation (4.54) one obtains resulting elastoplastic stiffness relation in the damaged configuration is obtained as follows:

$$\dot{\sigma} = D : \dot{\epsilon} \tag{4.67}$$

where

$$D = O^{-1} : \bar{D} : M^{-1} \tag{4.68}$$

where $\mathbf{O}$ is a fourth order tensor that can be derived based on the evolution equations for damage and is outlined in detail in [7] for the case of uniaxial loading. The effective stiffness tensor $\bar{D}$ maybe obtained from either a continuum approach or a micromechanical approach. These approaches are outlined below.
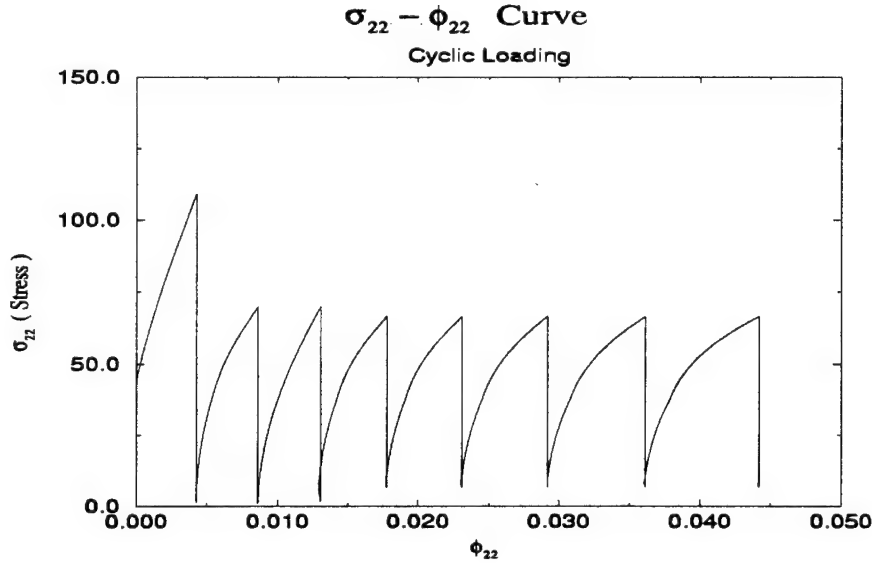
Figure 4.11: Evolution of Damage Parameter ϕ_{22} with Transverse Stress

4.5.1 Effective Stiffness Tensor $\bar{D}$ for Continuum Model

For the continuum-damage model the effective undamaged elasto-plastic relationship is given by the stiffness generated by the cyclic plasticity model. No modifications are made at this stage. $\bar{D}$ is the effective undamaged elasto-plastic stiffness and is given by,

$$\bar{D} = \bar{E} - \frac{(\bar{E} : \bar{n})(\bar{n} : \bar{E})}{H + (\bar{n} : \bar{E} : \bar{n})} \tag{4.69}$$

No other computations are necessary.

4.5.2   Effective Stiffness Tensor   for Micromechanical Model

The stiffness tensor D for the damaged material now derived for isothermal conditions and in the absence of rate dependent effects. Making use of the incremental form of equation (4.54) one obtains

$$\dot{\bar{\sigma}} = \dot{M} : \sigma + M : \dot{\sigma} \tag{4.70}$$

Through the additive decomposition of the effective strain rate one obtains

$$\dot{\bar{\epsilon}} = \dot{M}^{-1} : \epsilon^{-1} + M^{-1} : \dot{\epsilon} \tag{4.71}$$

Making use of equation (4.63) the rates of the damage effect tensor maybe expressed as follows

$$\dot{M}_{ijkl} = \frac{\partial M_{ijkl}}{\partial \phi_{pq}} T_{pqmn} \dot{\sigma} \tag{4.72}$$

$$= Q_{ijklmn} \dot{\sigma}_{mn} \tag{4.73}$$

and the inverse of M is given by,

$$\dot{M}_{ijkl}^{-1} = \frac{\partial M_{ijkl}^{-1}}{\partial \phi_{pq}} T_{pqmn}$$

$$= R_{ijklmn} \dot{\sigma}_{mn} \tag{4.74}$$

The elasto-plastic stiffness matrix in the undamaged configuration is given by equation (4.52). Making use of equations (4.71), (4.73), (4.74) and (4.52) the resulting elastoplastic stiffness relation in the damaged configuration is obtained as follows:

$$\dot{\sigma} = D : \dot{\epsilon} \tag{4.75}$$

where the damaged elasto-plastic stiffness is given by,

$$D = O^{-1} : \bar{D} : M^{-1} \tag{4.76}$$

and

$$O_{ijkl} = Q_{ijmnkl}\sigma_{mn} + M_{ijkl} - \bar{D}_{ijmn} R_{mnpqkl} E_{pqab} \tag{4.77}$$

In the above equation $\bar{D}$ is the effective undamaged elasto-plastic stiffness of the composite and can be expressed as,

$$\bar{D} = \bar{c}^m \bar{D}^m : \bar{A}^m + \bar{c}^f \bar{E}^f : \bar{A}^f \tag{4.78}$$

In the case of no damage, both tensors Q and R reduce to zero and M becomes a fourth order identity tensor.

Shear Stress–Strain Comparison

Damage/No Damage Cyclic Loading Models
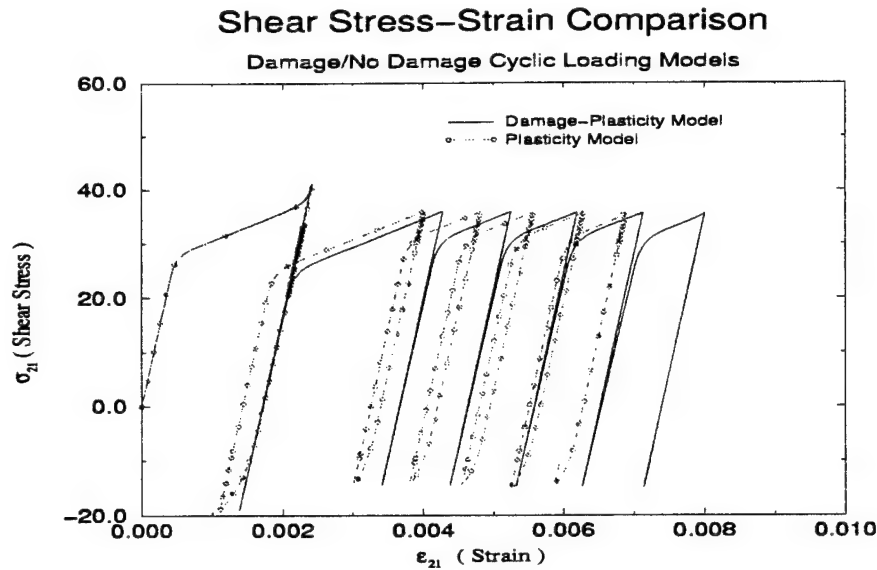
Figure 4.12: Shear Stress-Strain for Continuum Damage and Pure Plasticity Models

4.5.3 Continuum Damage Model Results

The same loading that was studied earlier, and used in the experimental work of Nigam et al.(1993) has been used here in this work. The damage parameters found suitable for this material were $q = 1.0$ and $r = 7.0$. This effectively makes it dependent only on one parameter. Figures 4.12, 4.14, 4.13, 4.15 show the results of this generation for the stress-strain comparison in the transverse and shear directions. These curves compare the cyclic-plasticity model with that of the damage-plasticity model. It can be seen that the strains predicted by the damage model are higher than that of the pure plasticity case. It can also be seen that during the unloading-reloading situation, when reloading takes place even in the elastic range, the damage criterion is exceeded, and hence the elastic-stiffness is reduced. This can be clearly seen in the two lines of different inclinations in Figures 4.13 and 4.15. As seen in Figures 4.13 and 4.15 which depict the stress-plastic strain relationships in the shear and transverse directions respectively, due to successive reduction in the elastic stiffness, the plastic strains are also affected hence resulting in a higher prediction of plastic strain. Although this model assumes a decoupling between damage and plasticity situations in modeling the behavior, there is an inherent coupling that is present.

Figure 4.11 shows the evolution of the damage parameter ϕ with stress in the transverse direction under a cyclic loading type of situation. One apparent behavior that is observed due to the nature of these curves is that as stress is increased, the same stress increment tends to produce a higher amount of damage. Upon unloading no significant change in damage is observed, and evolution of damage upon reloading takes place at a lower stress level for successive loading cases. Another behavior observed is that under constant load cycling, the amount of damage per cycle is higher as the number of times the load is applied increases. These behaviors observed are reasonable with what one would expect in reality.

The micromechanical model results are being developed and the results for them are

90

presented elsewhere.
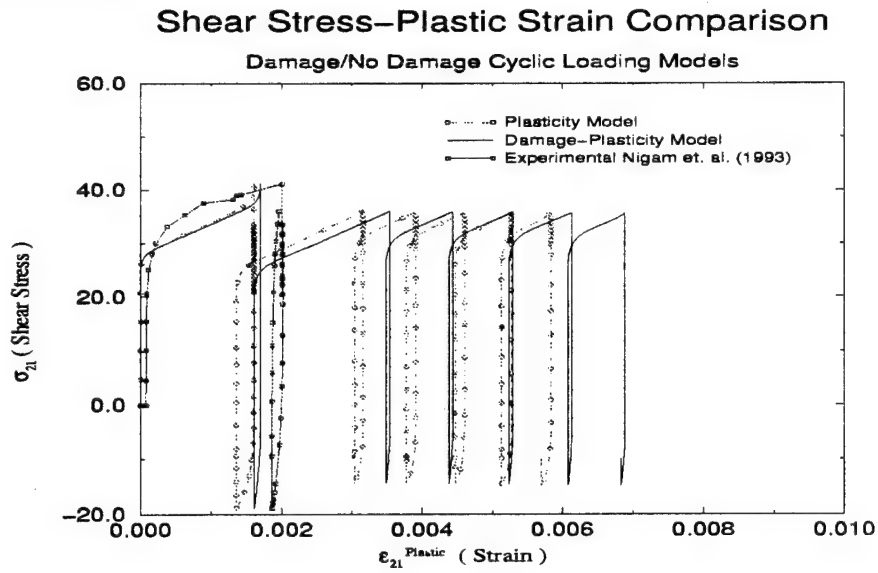
Figure 4.13: Shear Stress-Plastic Strain for Continuum Damage and Pure Plasticity Models

Transverse Stress–Strain Comparison

Damage/No Damage Cyclic Loading Models
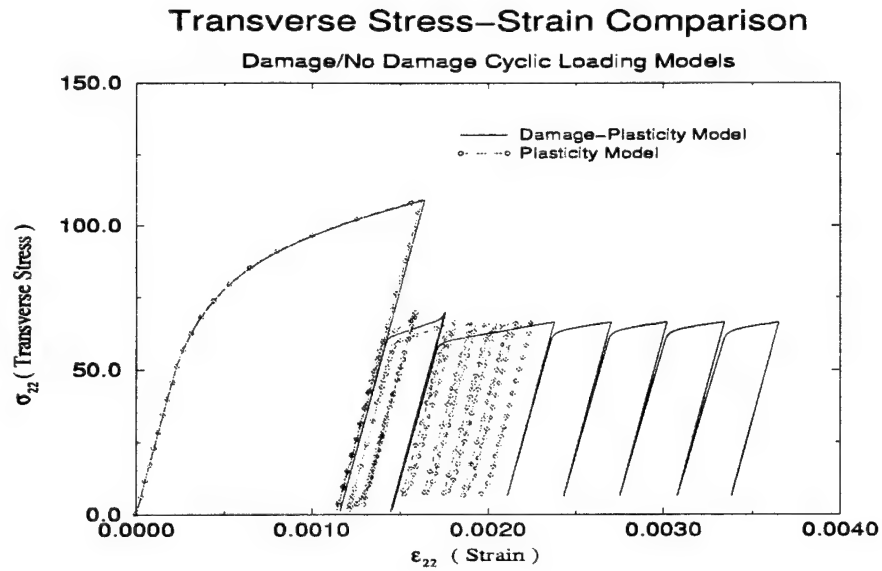
Figure 4.14: Transverse Stress-Strain for Continuum Damage and Pure Plasticity Models

Transverse Stress–Plastic Strain Comparison

Damage/No Damage Cyclic Loading Models
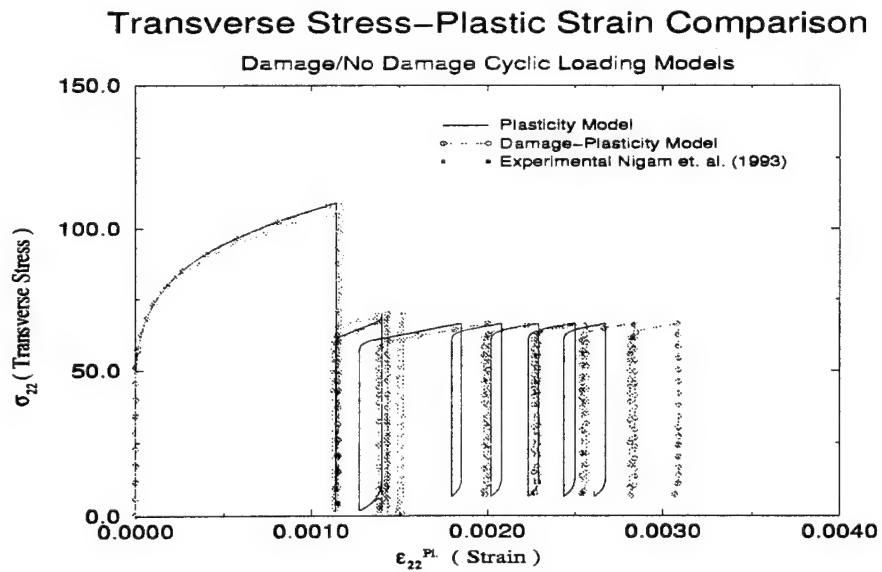
Figure 4.15: Transverse Stress-Plastic Strain for Continuum Damage and Pure Plasticity Models

Bibliography

[1] L. M. Kachanov. On the Creep Fracture Time. *Izv. Akad. Nauk. USSR Otd. Tekh.*, 8:26–31, 1958.

[2] G. Z. Voyiadjis and P. Kattan. A Continuum-Micromechanics Damage Model for Metal Matrix Composite. In *Composite Material Technology 1992,ASME,PD,Proceedings of the Composites Material Symposium of the energy Technology Conference and Exposition, Houston, Texas*, volume 45, pages 83–95, 1992.

[3] S. Murakami. Mechanical Modelling of Material Damage. *Journal of Applied Mechanics*, 55, 1988.

[4] Li G. Lee H. and Lee S. In *Int. Seminar on Local Approach of Fracture, Moret-sur-Loing, France*, pages 79–90, 1986.

[5] J. Lemaitre. Evaluation of Dissipation and Damage in Metals. *Proc. ICMI, Kyoto, Japan.*, 1971.

[6] G. Z. Voyiadjis and P. I. Kattan. Local Approach to Damage in Elasto-plastic Metal Matrix Composite. *Int. Journal of Damage Mechanics*, 2(1):92–114, 1993.

[7] G. Z. Voyiadjis and T. Park. Local and Interfacial Damage Analysis of Metal Matrix Composites. *International Journal of Engineering Science*, 33(11):1595–1621, 1995.

[8] P. I Kattan and G. Z. Voyiadjis. Micromechanical Analysis of Damage on Uniaxially Loaded Unidirectional Fiber-Reinforced Lamina. *Int. Journal of Solids and Structures*, 30(1):19–36, 1993.

[9] G. J. Dvorak and Bahei-El-Din. Elastic-Plastic Behavior of Fibrous Composite. *Journal of the Mechanics and Physics of Solids*, 27:51–72, 1979.

[10] G. Z. Voyiadjis and A. R. Venson. Experimental Damage Investigation of a SiC-Ti Aluminide Metal Matrix Composite. *International Journal of Damage Mechanics*, 41(4):338–361, October 1995.

[11] J. P. Cordebois and F. Sidoroff. Anisotropic Damage in Elasticity and Plasticity. *Journal de Mecanique Theorique et Appliquee, Numéro spécial*, pages 45–60, 1982.

[12] G. Z. Voyiadjis and P. I Kattan. A Plasticity-Damage Theory for Large Deformation of Solids. Part I: Theoretical Formulation. *International Journal of Engineering Science*, 30:1089–1108, 1992.

[13] T. Chen, George J. Dvorak, and Y. Benveniste. Mori-Tanaka Estimates of the Overall Elastic Moduli of Certain Composite Materials. *Journal of Applied Mechanics*, 59:539–546, September 1992.

[14] Y. A. Bahei-El-Din G. J. Dvorak. Elastic-Plastic Behavior of Fibrous Composites. *Journal of the Mechanics and Physics of Solids*, 27:51–72, 1979.

[15] G. Z. Voyiadjis. *Large Elasto-Plastic Deformation of Solids*. Ph.D. Dissertation, Dept. of Civil Engineering and Engineeing Mechanics, Columbia University, New York, NY, 1973.

[16] K. Bathe. *Finite Element Procedures in Engineering Analysis*. Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1990.

[17] G. Z. Voyiadjis and Ganesh Thiagarajan. An Anisotropic Yield Surface Model for Directionally Reinforced Metal Matrix Composites. *International Journal of Plasticity*, 110:151–172, 1995.

[18] Ganesh Thiagarajan. *A Cyclic Plasticity/Damage Model for Metal Matrix Composites*. Louisiana State University, 1996. Ph.D Dissertation.

[19] Y. F. Dafalias and E. P. Popov. Plastic Internal Variables Formalism of Cyclic Plasticity. *Journal of Applied Mechanics*, 43:645–651, 1976.

[20] H. Nigam, G. J. Dvorak, and Y. A. Bahei el Din. An Experimental Investigation of Elastic-Plastic Behavior of a Fibrous Boron-Aluminum Composite. I. Matrix-Dominated Mode. *International Journal of Plasticity*, 10:23–48, 1993.

[21] G. J. Dvorak, Y. A. Bahei el Din, Macheret Y., and Liu C. H. An Experimental Study of Elastic-Plastic Behavior of a Fibrous Boron-Aluminum Compisite. *Journal of the Mechanics and Physics of Solids*, 36:655–687, 1988.

[22] G. Z. Voyiadjis and P. I. Kattan. Local Approach to Damage in Elasto-Plastic Metal Matrix Composites. *Int. Journal of Damage Mechanics*, 2(1):92–114, 1993.

[23] G. J. Weng. The Overall Elastoplastic Stress-Strain Relations of Dual Phase Metals. *Journal of the Mechanics and Physics of Solids*, 38(3):419–441, 1990.

[24] R. Hill. A Theory of the Yielding and Plastic Flow of Anisotropic Metals. *Proceedings of Royal Society of London*, A193:281–297, 1948.

[25] J. F. Mulhern, T. G. Rogers, and A. J. M. Spencer. A Continuum Theory of an Elastic-Plastic Fiber Reinforced Material. *Proceedings of Royal Society of London*, 7:129–152, 1969.

[26] R. Hill. Continuum micro-mechanics of elasto-plastic polycrystals. *Journal of the Mechanics and Physics of Solids*, 13:89–101, 1965.

[27] A. J. M. Spencer. *Deformations of Fiber-reinforced Materials*. Clarendon Press, Oxford, 1972.

[28] L. J. Walpole. On the Overall Elastic Moduli of Composite Materials. *Journal of the Mechanics of Physics and Solids*, 17:235–251, 1969.

[29] J. F. Mulhern, T. G. Rogers, and A. J. M. Spencer. A Continuum Model for a Fiber Reinforced Plastic Material. *Proceedings of Royal Society of London*, A301:473–492, 1967.

[30] G. Z. Voyiadjis and Ganesh Thiagarajan. A Cyclic Plasticity Model for Metal Matrix Composites. Accepted for publication, 1996.

[31] J. L. Teply and George J. Dvorak. Bounds on Overall Instantaneous Properties f Elastic-Plastic Composites. *Journal of the Mechanics and Physics of Solids*, 36(1):29–58, 1988.

[32] B. Budiansky. On the Elastic Moduli of Some Heterogeneous Materials. *Journal of Mechanics of Physics and Solids*, 13:223–227, 1965.

[33] D. C. Lagoudas, A. C. Gavazzi, and H. Nigam. Elastoplastic Behavior of Metal Matrix Composites Based on Incremental Plasticity and the Mori-Tanaka Averaging Scheme. *Computational Mechanics*, 8:193–203, 1991.

[34] M. Stumvoll and G. Swoboda. Deformation Behavior of Ductile Solids Containing Anisotropic Damage. *Journal of Engineering Mechanics*, 119(7):169–192, July 1993.

Appendix

FORTRAN CODING OF SUBROUTINES
TO CALCULATE DAMAGE ELASO-PLASTIC
STIFFNESS D_{ijkl}

```
C =================================================================
C ====================== M A T M O D ==============================
C =================================================================
C
      SUBROUTINE MATMOD(ELNUM,INTGPN,MATNUM,ICODE,ICRK)
      IMPLICIT REAL*8 (A-H,O-Z)
      INTEGER ELNUM
      CHARACTER*1 ICRK
      COMMON/INPUTF/MATYPE(10)
C
      I=MATYPE(MATNUM)
      IF (I.EQ.1) THEN
        CALL ODCOM(ICODE,ELNUM,INTGPN,ICRK)
      ELSE IF(I.EQ.2) THEN
        CALL LDCOM(ICODE,ELNUM,INTGPN,ICRK)
      ELSE
        WRITE (IOUT,100) I
        STOP
      END IF
C
      RETURN
  100 FORMAT (/1X,'INVALID MATERIAL TYPE(',I3,') SPECIFIED')
      END
C
C =================================================================
C ===========          S T A R T   O F         =========
C ===========  E L A S T O - P L A S T I C   M O D E L  =========
C ===========  W I T H   D A M A G E   M E C H A N I C S  =========
C ===========      F O R   T H E   O R T H O T R O P I C  =========
C ===========        C O P O S I T E   L A M I N A T E  =========
C ===========  U S I N G   O V E R A L L   A P P R O A C H  =========
C =================================================================
C ======================= O D C O M  ==============================
C =================================================================
      SUBROUTINE ODCOM(ICODE,ELNUM,INTGPN,ICRK)
      INTEGER ELNUM
      CHARACTER*1 ICRK
C
      IF (ICODE.EQ.0) THEN
        CALL OCALSTF(ELNUM,INTGPN)
      ELSE
        CALL OSTRDAM(ELNUM,INTGPN,ICRK)
      END IF
C
      RETURN
      END
C =================================================================
C ====================== O M E D A M ==============================
C =================================================================
      SUBROUTINE OMEDAM
C =================================================================
C I                                                               I
C I  P R O G R A M:                                               I
C I                                                               I
C I    PROGRAM 'MEDAM' IS THE CONTROL UNIT FOR CALCULATION OF THE I
C I    ELASTIC STRESS-STRAIN STIFFNESS MATRIX INCLUDING THE       I
C I    EFFECT OF DAMAGE.                                          I
C I                                                               I
C =================================================================
      IMPLICIT REAL*8 (A-H,O-Z)
      INTEGER ELNUM
      CHARACTER*48 CSTRN,CSTRS,CASTRS,CSTRM,CPHI,CDPHI,CCENT
      CHARACTER*8 CHK,CYY
      CHARACTER*1 CDMG,ICRK
      COMMON/DEVICE/LDEV1,LDEV2,LDEV3,LDEV4,LDEV,LDEVST
      COMMON/LAYTP1/NP,NTP
      COMMON/ELSTR2/STRS(6)
      COMMON/MATER1/DEP(6,6)
      COMMON/INPUT1/THICK,NPLY
      COMMON/INPUT2/MATL(10),DMGPRM(27)
      COMMON/INPUT3/DEGRE(10),PLYTHK(10)
      COMMON/CONTR1/INCREM,NIT
C
      COMMON/MEDAM1/ESMB(6,6),ESFB(6,6)
      COMMON/MEDAM2/ECMB(6,6),ECFB(6,6)
      COMMON/AFECON1/AFE(6,6)
      COMMON/AMECON1/AME(6,6)
      COMMON/BFECON1/BFE(6,6)
      COMMON/BMECON1/BME(6,6)
C
```

```
      DIMENSION STRAIN(6),DE(6),TDE(6)
C
      DIMENSION STRESS(6),TSTRS(6),DS(6),TDS(6)
      DIMENSION AVGSTR(6),BTDS(6)
C
      DIMENSION STRSM(6)
      DIMENSION DSM(6),DSTRSM(6)
C
      DIMENSION EPLY(6,6),EPLYB(6,6),DEPLY(6,6),GRS(6,6)
      DIMENSION ECB(6,6),DPLY(6,6),PSMB(6,6)
      DIMENSION DUMMY(6),DUMMY1(6),DUMMY2(6)
C
      DIMENSION FS(6),CENT(6)
C
      DIMENSION PHI(6),DPHI(6),DM(6,6),DMP(6,6,6)
      DIMENSION Y(6),HI(6)
C
      DIMENSION NCRACK(5)
C
      DIMENSION DDSM(6),TSTRSM(6),DDS(6),TDDS(6),DDE(6),TDDE(6)
      DIMENSION BTDDS(6),DDPHI(6)
C
      EQUIVALENCE (CSTRN,STRAIN),(CASTRS,AVGSTR),(CSTRM,STRSM),
     $(CPHI,PHI),(CDPHI,DPHI),(CHK,HK),(CSTRS,STRESS),(CYY,YY)
C
C ===================ENTRY OSTRDAM =================================
C
      ENTRY OSTRDAM(ELNUM,INTGPN,ICRK)
C
      IF (INCREM.GT.1) THEN
        READ(LDEV1,11) CSTRN
      ELSE
        DO 10 I=1,6
   10   STRAIN(I)=0.D0
      END IF
C
C --- CALCULATION OF THE STRAIN INCREMENT FOR LAMINATE
C
      CALL CALSTRN(DE,STRAIN)
C
      WRITE(LDEV2,11) CSTRN
C
C --- GET THE MATERIAL PARAMETERS
C
      CMO=MATL(1)
      CFO=MATL(2)
      EM=MATL(3)
      EF=MATL(4)
      UM=MATL(5)
      UF=MATL(6)
      GM=EM/(2.D0*(1.D0+UM))
      GF=EF/(2.D0*(1.D0+UF))
C
C --- GET THE YIELD PARAMETERS
C
      SY=MATL(7)
      B=MATL(8)
C
C --- GET THE DAMAGE PARAMETERS
C
      R1=DMGPRM(1)
      R2=DMGPRM(2)
      R3=DMGPRM(3)
      Q1=DMGPRM(4)
      Q2=DMGPRM(5)
      Q3=DMGPRM(6)
      V1=DMGPRM(7)
      V2=DMGPRM(8)
      V3=DMGPRM(9)
C
C ------------------------------------------------------------------------
C | CALCULATION OF THE STRESS INCREMENT & TOTAL STRESS ON EACH PLY    |
C ------------------------------------------------------------------------
C   TYPE OF LAMINATE
C     NTP=1 ; SINGLE LAMINAE
C     NTP=2 ; EVEN NUMBER OF LAMINAE
C     NTP=3 ; ODD NUMBER OF LAMINAE
C ------------------------------------------------------------------------
C
      DO 100 KP=1,NP
```

98

```
C
      IF (INCREM.GT.1) THEN
        READ(LDEV1,22) CASTRS,CYY,CPHI,CDPHI,CSTRM,CCENT,CHK,CDMG
      ELSE
        DO 110 I=1,6
        AVGSTR(I)=0.D0
        PHI(I)=0.D0
        DPHI(I)=0.D0
        CENT(I)=0.D0
  110   STRSM(I)=0.D0
        HK=0.D0
        YY=0.D0
        CDMG=' '
      END IF
C
      DO 120 I=1,6
  120 TSTRSM(I)=STRSM(I)
C
C --- GET THE FIBER DIRECTION OF PLY AND THICKNESS
C
      THETA=DEGRE(KP)
      THKK=PLYTHK(KP)
      SCALE=10.D0
C
C --- COMPUTATION A TRIAL ELASTIC STRESS OF MATRIX
C
      CALL AECON(CMO,CFO,EM,GM,UM)
      CALL BECON(CMO,CFO)
      CALL PLYSTF(EPLY,ESMB,ESFB,CMO,CFO)
      IF (CDMG.EQ.'Y') THEN
        CALL DMAGE2(DM,DMP,PHI)
        CALL ELDAMG(DEPLY,DM,EPLY)
        CALL TRNSTF(DEPLY,THETA)
        CALL CONVER(GRS,DEPLY)
      ELSE
        CALL TRNSTF(EPLY,THETA)
        CALL CONVER(GRS,EPLY)
      END IF
      CALL CALSTRS(DUMMY1,DUMMY2,DS,GRS,DE)
      CALL TRNSTR(TDS,DS,THETA)
      CALL TRNSTR(DUMMY1,DUMMY2,THETA)
      IF (CDMG.EQ.'Y') THEN
        CALL EFECT3(BTDS,TDS,DMP,DM,DPHI,DUMMY1)
        CALL LOSTRS(DSM,BME,BTDS)
      ELSE
        CALL LOSTRS(DSM,BME,TDS)
      END IF
      CALL TRNSTN(TDE,DE,THETA)
      CALL LOSTRS(DSTRNM,AME,TDE)
C
      CALL UPDATE(TSTRSM,DSM)
C
C-----
      IF (CDMG.EQ.' ') THEN
C-----
      IF (YY.GT.0.9) THEN
        CALL AECON(CMO,CFO,EM,GM,UM)
        CALL BECON(CMO,CFO)
        CALL ELPLD(PSMB,ESMB,STRSM,CENT,B)
        CALL APCON(PSMB,CMO,CFO,EM,EM,GM,GM,UM)
        CALL BPCON(PSMB,CMO,CFO)
        CALL PLYSTF(EPLY,PSMB,ESFB,CMO,CFO)
        CALL TRNSTF(EPLY,THETA)
        CALL CONVER(GRS,EPLY)
        CALL CALSTRS(DUMMY,AVGSTR,DS,GRS,DE)
        CALL TRNSTR(TDS,DS,THETA)
        CALL TRNSTR(TSTRS,AVGSTR,THETA)
        CALL LOSTRS(DSM,BME,TDS)
        CALL UPDATE(STRSM,DSM)
        CALL GRSSTF(DEP,GRS,NTP,KP,NP,THKK,THICK)
      ELSE
        CALL YIELD(FY,TSTRSM,CENT,SY)
        IF (FY.LT.0.0) THEN
          CALL AECON(CMO,CFO,EM,GM,UM)
          CALL BECON(CMO,CFO)
          CALL PLYSTF(EPLY,ESMB,ESFB,CMO,CFO)
          CALL TRNSTF(EPLY,THETA)
          CALL CONVER(GRS,EPLY)
          CALL CALSTRS(DUMMY,AVGSTR,DS,GRS,DE)
          CALL TRNSTR(TDS,DS,THETA)
```

99

```
              CALL TRISTR(TSTRS,AVGSTR,THETA)
              CALL LOSTRS(DSM,BME,TDS)
              CALL UPDATE(STRSM,DSM)
              CALL GRSSTF(DEP,GRS,ITP,KP,IP,THKK,THICK)
            ELSE
              YY=1.D0
              ISBI=IIT(SCALE+0.1)
              DO 125 L=1,6
 125          DDE(L)=DE(L)/SCALE
              CALL AECOI(CMO,CFO,EM,GM,UM)
              CALL BECOI(CMO,CFO)
              CALL PLYSTF(EPLY,ESMB,ESFB,CMO,CFO)
              CALL TRISTF(EPLY,THETA)
              CALL COIVER(GRS,EPLY)
              CALL CALSTRS(DUMMY1,AVGSTR,DDS,GRS,DDE)
              CALL TRISTR(TDDS,DDS,THETA)
              CALL TRISTR(TSTRS,AVGSTR,THETA)
              CALL LOSTRS(DDSM,BME,TDDS)
              CALL UPDATE(STRSM,DDSM)
              DO 130 I=1,ISBI-1
              IF (IYY.EQ.10) THEI
                FY=1.0
              ELSE
                CALL YIELD(FY,STRSM,CEIT,SY)
              EID IF
              IF (FY.LT.0.0) THEI
              CALL AECOI(CMO,CFO,EM,GM,UM)
              CALL BECOI(CMO,CFO)
              CALL PLYSTF(EPLY,ESMB,ESFB,CMO,CFO)
              CALL TRISTF(EPLY,THETA)
              CALL COIVER(GRS,EPLY)
              CALL CALSTRS(DUMMY,AVGSTR,DDS,GRS,DDE)
              CALL TRISTR(TDDS,DDS,THETA)
              CALL TRISTR(TSTRS,AVGSTR,THETA)
              CALL LOSTRS(DDSM,BME,TDDS)
              CALL UPDATE(STRSM,DDSM)
              ELSE
              IYY=10
              CALL AECOI(CMO,CFO,EM,GM,UM)
              CALL BECOI(CMO,CFO)
              CALL ELPLD(PSMB,ESMB,STRSM,CEIT,B)
              CALL APCOI(PSMB,CMO,CFO,EM,EM,GM,GM,UM)
              CALL BPCOI(PSMB,CMO,CFO)
              CALL PLYSTF(EPLY,PSMB,ESFB,CMO,CFO)
              CALL TRISTF(EPLY,THETA)
              CALL COIVER(GRS,EPLY)
              CALL CALSTRS(DUMMY1,AVGSTR,DDS,GRS,DDE)
              CALL TRISTR(TDDS,DDS,THETA)
              CALL TRISTR(TSTRS,AVGSTR,THETA)
              CALL LOSTRS(DDSM,BME,TDDS)
              CALL UPDATE(STRSM,DDSM)
              EID IF
 130          COITIIUE
              CALL GRSSTF(DEP,GRS,ITP,KP,IP,THKK,THICK)
            EID IF
          EID IF
C-----
          ELSE
C-----
C
          CALL DMAGE2(DM,DMP,PHI)
          IF (YY.GT.0.9) THEI
            CALL AECOI(CMO,CFO,EM,GM,UM)
            CALL BECOI(CMO,CFO)
            CALL ELPLD(PSMB,ESMB,STRSM,CEIT,B)
            CALL APCOI(PSMB,CMO,CFO,EM,EM,GM,GM,UM)
            CALL BPCOI(PSMB,CMO,CFO)
            CALL PLYSTF(EPLY,PSMB,ESFB,CMO,CFO)
            CALL ELDAMG(DEPLY,DM,EPLY)
            CALL DELPLD(DPLY,DEPLY,ECB,DM,DMP,STTRS)
            CALL TRISTF(DPLY,THETA)
            CALL COIVER(GRS,DPLY)
            CALL CALSTRS(DUMMY,AVGSTR,DS,GRS,DE)
            CALL TRISTR(TDS,DS,THETA)
            CALL TRISTR(TSTRS,AVGSTR,THETA)
            CALL EFECT3(BTDS,TDS,DMP,DM,DPHI,DUMMY1)
            CALL LOSTRS(DSM,BME,BTDS)
            CALL UPDATE(STRSM,DSM)
            CALL GRSSTF(DEP,GRS,ITP,KP,IP,THKK,THICK)
          ELSE
```

100

```
          CALL YIELD(FY,TSTRSM,CENT,SY)
          IF (FY.LT.0.0) THEN
            CALL AECON(CMO,CFO,EM,GM,UM)
            CALL BECON(CMO,CFO)
            CALL PLYSTF(EPLY,ESMB,ESFB,CMO,CFO)
            CALL ELDAMG(DEPLY,DM,EPLY)
            CALL TRNSTF(DEPLY,THETA)
            CALL CONVER(GRS,DEPLY)
            CALL CALSTRS(DUMMY,AVGSTR,DS,GRS,DE)
            CALL TRNSTR(TDS,DS,THETA)
            CALL TRNSTR(TSTRS,AVGSTR,THETA)
            CALL EFECT3(BTDS,TDS,DMP,DM,DPHI,DUMMY1)
            CALL LOSTRS(DSM,BME,BTDS)
            CALL UPDATE(STRSM,DSM)
            CALL GRSSTF(DEP,GRS,NTP,KP,NP,THKK,THICK)
          ELSE
            YY=1.DO
            NSBI=INT(SCALE+0.1)
            DO 135 L=1,6
135         DDE(L)=DE(L)/SCALE
            CALL AECON(CMO,CFO,EM,GM,UM)
            CALL BECON(CMO,CFO)
            CALL PLYSTF(EPLY,ESMB,ESFB,CMO,CFO)
            CALL ELDAMG(DEPLY,DM,EPLY)
            CALL TRNSTF(DEPLY,THETA)
            CALL CONVER(GRS,DEPLY)
            CALL CALSTRS(DUMMY1,AVGSTR,DDS,GRS,DDE)
            CALL TRNSTR(TDDS,DDS,THETA)
            CALL TRNSTR(TSTRS,AVGSTR,THETA)
            CALL EFECT3(BTDDS,TDDS,DMP,DM,DPHI,DUMMY1)
            CALL LOSTRS(DDSM,BME,BTDDS)
            CALL UPDATE(STRSM,DDSM)
            DO 140 I=1,NSBI-1
            IF (IYY.EQ.10) THEN
              FY=1.0
            ELSE
              CALL YIELD(FY,STRSM,CENT,SY)
            END IF
            IF (FY.LT.0.0) THEN
            CALL AECON(CMO,CFO,EM,GM,UM)
            CALL BECON(CMO,CFO)
            CALL PLYSTF(EPLY,ESMB,ESFB,CMO,CFO)
            CALL ELDAMG(DEPLY,DM,EPLY)
            CALL TRNSTF(DEPLY,THETA)
            CALL CONVER(GRS,DEPLY)
            CALL CALSTRS(DUMMY,AVGSTR,DDS,GRS,DDE)
            CALL TRNSTR(TDDS,DDS,THETA)
            CALL TRNSTR(TSTRS,AVGSTR,THETA)
            CALL EFECT3(BTDDS,TDDS,DMP,DM,DPHI,DUMMY1)
            CALL LOSTRS(DDSM,BME,BTDDS)
            CALL UPDATE(STRSM,DDSM)
            ELSE
            IYY=10
            CALL AECON(CMO,CFO,EM,GM,UM)
            CALL BECON(CMO,CFO)
            CALL ELPLD(PSMB,ESMB,STRSM,CENT,B)
            CALL APCON(PSMB,CMO,CFO,EM,EM,GM,GM,UM)
            CALL BPCON(PSMB,CMO,CFO)
            CALL PLYSTF(EPLY,PSMB,ESFB,CMO,CFO)
            CALL ELDAMG(DEPLY,DM,EPLY)
            CALL DELPLD(DPLY,DEPLY,ECB,DM,DMP,TSTRS)
            CALL TRNSTF(DPLY,THETA)
            CALL CONVER(GRS,DPLY)
            CALL CALSTRS(DUMMY1,AVGSTR,DDS,GRS,DDE)
            CALL TRNSTR(TDDS,DDS,THETA)
            CALL TRNSTR(TSTRS,AVGSTR,THETA)
            CALL EFECT3(BTDDS,TDDS,DMP,DM,DPHI,DUMMY1)
            CALL LOSTRS(DDSM,BME,BTDDS)
            CALL UPDATE(STRSM,DDSM)
            END IF
140         CONTINUE
            CALL GRSSTF(DEP,GRS,NTP,KP,NP,THKK,THICK)
          END IF
        END IF
C-----
        END IF
C-----
C
C ---------------------------------------------------------------
C |                        UPDATE BACK-STRESS                          |
```

101

```fortran
C --------------------------------------------------------------------
C
      IF (YY.GT.0.9) THEN
        CALL FDER(FS,STRSM,CENT)
        CALL QSCALAR(Q,ESMB,STRSM,CENT,B)
        CALL LAMUDA(ALAM,ESMB,FS,DSTRMM,Q)
        CALL USCALAR(UT,STRSM,CENT,ALAM,B)
        CALL CENTER(CENT,STRSM,UT)
      END IF
C
C --------------------------------------------------------------------
C |  UPDATE DMAGE ON EACH PLY                                        |
C --------------------------------------------------------------------
C
      IF(CDMG.NE.'F') THEN
C
      CALL DMAGE2(DM,DMP,PHI)
      CALL DMGFRC(Y,DM,DMP,TSTRS,ECB)
C
      U1=EPLYB(1,1)-2.D0*EPLYB(4,4)
      U2=EPLYB(2,2)-2.D0*EPLYB(5,5)
      U3=EPLYB(3,3)-2.D0*EPLYB(6,6)
C
      CALL DMGHAR(HI,A1,A2,A3,B1,B2,B3,PHI,HK,R1,R2,R3,Q1,Q2,Q3
     $,U1,U2,U3,V1,V2,V3)
C
      CALL DMGCRT(GCRT,Y,HI)
C
      IF (GCRT.GE.1.D0) THEN
       CDMG='Y'
       CALL CALDMG(PHI,DPHI,TSTRS,TDS,ECB,HK,Y,HI,A1,A2,A3,B1,B2,B3
     $,CDMG,1)
      END IF
      END IF
C
      IF (CDMG.EQ.'F') THEN
        DO 515 J=1,3
 515    PHI(J)=0.90
        NCRACK(KP)=1
        WRITE(6,777) ELNUM,INTGPN,KP,THETA
 777  FORMAT(2X,'LAMINA CRACK',2X,I5,2X,I2,2X,I2,2X,F4.1)
      END IF
C
      WRITE(LDEV2,22) CASTRS,CYY,CPHI,CDPHI,CSTRM,CCENT,CHK,CDMG
C
 100  CONTINUE
C
C --------------------------------------------------------------------
C |  UPDATE AVERAGE STRESSES  OF LAMINATE                            |
C --------------------------------------------------------------------
C
      IF (INCREM.GT.1) THEN
        READ(LDEV1,11) CSTRS
      ELSE
        DO 200 K1=1,6
 200    STRESS(K1)=0.D0
      END IF
C
      CALL CALSTRS(STRS,STRESS,DUMMY,DEP,DE)
C
      WRITE(LDEV2,11) CSTRS
C
      CALL CHKCRK(NP,NCRACK,ICRK)
C
      IF (ICRK.EQ.'Y') THEN
        WRITE(6,878) ELNUM,INTGPN
 878    FORMAT(2X,'LAMINATE CRACK',2X,I5,2X,I2)
      END IF
C
      RETURN
C ====================================================================
C ==================== ENTRY OCALSTF  ================================
C ====================================================================
      ENTRY OCALSTF(ELNUM,INTGPN)
C
      IF (INCREM.GT.1) THEN
        IF (NIT.EQ.1) THEN
          READ(LDEV1,11) CSTRN
        ELSE
          READ(LDEV2,11) CSTRN
```

```
          END IF
        END IF
C
C --- GET THE MATERIAL PARAMETERS
C
      CMO=MATL(1)
      CFO=MATL(2)
      EM=MATL(3)
      EF=MATL(4)
      UM=MATL(5)
      UF=MATL(6)
      GM=EM/(2.DO*(1.DO+UM))
      GF=EF/(2.DO*(1.DO+UF))
C
C --- GET THE YIELD PARAMETERS
C
      B=90.DO
C
C --- COMPUTE ELASTIC CONSTANTS
C
      CALL ADMAT(ESMB,EM,UM,GM)
      CALL ADMAT(ESFB,EF,UF,GF)
C
C -----------------------------------------------------------------------
C    TYPE OF LAMINATE
C      NTP=1 ; SINGLE LAMINAE
C      NTP=2 ; EVEN NUMBER OF LAMINAE
C      NTP=3 ; ODD NUMBER OF LAMINAE
C -----------------------------------------------------------------------
C
      DO 300 LP=1,NP
C
      IF (INCREM.GT.1) THEN
      IF (NIT.EQ.1) THEN
       READ(LDEV1,22) CASTRS,CYY,CPHI,CDPHI,CSTRM,CCENT,CHK,CDMG
      ELSE
       READ(LDEV2,22) CASTRS,CYY,CPHI,CDPHI,CSTRM,CCENT,CHK,CDMG
      END IF
      ELSE
       CDMG=' '
       YY=0.DO
      END IF
C
C --- GET THE FIBER DIRECTION OF PLY AND THICKNESS
C
      THETA=DEGRE(LP)
      THKK=PLYTHK(LP)
C
      CALL AECON(CMO,CFO,EM,GM,UM)
C-----
      IF (CDMG.NE.'Y') THEN
C-----
      IF (YY.GT.0.9) THEN
        CALL ELPLD(PSMB,ESMB,STRSM,CENT,B)
        CALL APCON(PSMB,CMO,CFO,EM,EM,GM,GM,UM)
        CALL PLYSTF(EPLY,PSMB,ESFB,CMO,CFO)
        CALL TRNSTF(EPLY,THETA)
        CALL CONVER(GRS,EPLY)
        CALL GRSSTF(DEP,GRS,NTP,LP,NP,THKK,THICK)
      ELSE
        CALL PLYSTF(EPLY,ESMB,ESFB,CMO,CFO)
        CALL TRNSTF(EPLY,THETA)
        CALL CONVER(GRS,EPLY)
        CALL GRSSTF(DEP,GRS,NTP,LP,NP,THKK,THICK)
      END IF
C-----
      ELSE
C-----
C
      CALL DMAGE1(DM,PHI)
      IF (YY.GT.0.9) THEN
        CALL ELPLD(PSMB,ESMB,STRSM,CENT,B)
        CALL APCON(PSMB,CMO,CFO,EM,EM,GM,GM,UM)
        CALL PLYSTF(EPLY,ESMB,ESFB,CMO,CFO)
        CALL ELDAMG(DEPLY,DM,EPLY)
        CALL TRNSTF(DEPLY,THETA)
        CALL CONVER(GRS,DEPLY)
        CALL GRSSTF(DEP,GRS,NTP,LP,NP,THKK,THICK)
      ELSE
        CALL PLYSTF(EPLY,ESMB,ESFB,CMO,CFO)
```

103

```
            CALL ELDAMG(DEPLY,DM,EPLY)
            CALL TRNSTF(DEPLY,THETA)
            CALL CONVER(GRS,DEPLY)
            CALL GRSSTF(DEP,GRS,NTP,LP,NP,THKK,THICK)
         END IF
C-----
      END IF
C-----
C
 300  CONTINUE
C
      IF (INCREM.GT.1) THEN
         IF (NIT.EQ.1) THEN
           READ(LDEV1,11) CSTRS
         ELSE
           READ(LDEV2,11) CSTRS
         END IF
         BACKSPACE(UNIT=LDEV)
         BACKSPACE(UNIT=LDEV)
         BACKSPACE(UNIT=LDEV)
         BACKSPACE(UNIT=LDEV)
      END IF
C
      RETURN
C
 11   FORMAT(A48)
 22   FORMAT(A48,A8,4A48,A8,A1)
      END
C
C ======================================================================
C ===========              S T A R T   O F          =========
C =========== E L A S T O - P L A S T I C    M O D E L  =========
C =========== W I T H   D A M A G E   M E C H A N I C S  =========
C ===========    F O R   T H E   O R T H O T R O P I C   =========
C ===========        C O P O S I T E   L A M I N A T E   =========
C =========== U S I N G   L O C A L L   A P P R O A C H  =========
C ======================================================================
C ======================== L D C O M  ==================================
C ======================================================================
      SUBROUTINE LDCOM(ICODE,ELNUM,INTGPN,ICRK)
      INTEGER ELNUM
      CHARACTER*1 ICRK
C
      IF (ICODE.EQ.0) THEN
        CALL LCALSTF(ELNUM,INTGPN)
      ELSE
        CALL LSTRDAM(ELNUM,INTGPN,ICRK)
      END IF
C
      RETURN
      END
C ======================================================================
C ======================== L M E D A M =================================
C ======================================================================
      SUBROUTINE LMEDAM
C ======================================================================
C I                                                                  I
C I  P R O G R A M:                                                  I
C I                                                                  I
C I     PROGRAM 'MEDAM' IS THE CONTROL UNIT FOR CALCULATION OF THE   I
C I     ELASTIC STRESS-STRAIN STIFFNESS MATRIX INCLUDING THE         I
C I     EFFECT OF DAMAGE.                                            I
C I                                                                  I
C ======================================================================
      IMPLICIT REAL*8 (A-H,O-Z)
      INTEGER ELNUM
      CHARACTER*288 CDM
      CHARACTER*48 CSTRN,CSTRS,CAST,CSTM,CPM,CPF,CPB,CSTF,CCENT
      CHARACTER*8 CHKM,CHKF,CHKB,CYY
      CHARACTER*1 CDMG,ICRK
      COMMON/DEVICE/LDEV1,LDEV2,LDEV3,LDEV4,LDEV,LDEVST
      COMMON/LAYTP1/NP,NTP
      COMMON/ELSTR2/STRS(6)
      COMMON/MATER1/DEP(6,6)
      COMMON/INPUT1/THICK,NPLY
      COMMON/INPUT2/MATL(10),DMGPRM(27)
      COMMON/INPUT3/DEGRE(10),PLYTHK(10)
      COMMON/CONTR1/INCREM,NIT
C
      COMMON/MEDAM1/ESMB(6,6),ESFB(6,6)
```

104

```fortran
      COMMON/MEDAM2/ECMB(6,6),ECFB(6,6)
      COMMON/AFECON1/AFE(6,6)
      COMMON/AMECON1/AME(6,6)
      COMMON/BFECON1/BFE(6,6)
      COMMON/BMECON1/BME(6,6)
C
      DIMENSION ESM(6,6),ESF(6,6),PSMB(6,6)
      DIMENSION EPLYB(6,6),EPLY(6,6),DEPLY(6,6),ECB(6,6),GRS(6,6)
C
      DIMENSION STRAIN(6),DE(6),TDE(6),DDE(6),TDDE(6)
C
      DIMENSION STRESS(6),TSTRS(6),DS(6),TDS(6),DDS(6),TDDS(6)
      DIMENSION AVGSTR(6)
C
      DIMENSION DUMMY(6),DUMMY1(6),DUMMY2(6)
C
      DIMENSION FS(6),CENT(6)
C
      DIMENSION YM(6),YF(6),YB(6),HMI(6),HFI(6),HBI(6)
      DIMENSION PHIM(6),PHIF(6),PHIB(6)
      DIMENSION DM(6,6),DMM(6,6),DMF(6,6),DMB(6,6)
      DIMENSION DMPM(6,6,6),DMPF(6,6,6),DMPB(6,6,6)
C
      DIMENSION STRSM(6),STRSF(6),DSM(6),DSF(6),DDSM(6),DDSF(6)
      DIMENSION DSTRNM(6)
      DIMENSION BSTRSM(6),BDSTRNM(6)
      DIMENSION TSTRSM(6),BTSTRSM(6)
C
      DIMENSION NCRACK(5)
C
      EQUIVALENCE (CSTRN,STRAIN),(CAST,AVGSTR),(CSTM,STRSM),
     $(CSTF,STRSF),(CPM,PHIM),(CPF,PHIF),(CPB,PHIB),
     $(CHKM,HKM),(CHKF,HKF),(CHKB,HKB),(CYY,YY),(CDM,DM),
     $(CSTRS,STRESS)
C
C
C ====================ENTRY LSTRDAM ===============================
C
      ENTRY LSTRDAM(ELNUM,INTGPN,ICRK)
C
      IF (INCREM.GT.1) THEN
        READ(LDEV1,11) CSTRN
      ELSE
        DO 10 I=1,6
  10    STRAIN(I)=0.DO
      END IF
C
C --- CALCULATION OF THE STRAIN INCREMENT FOR LAMINATE
C
      CALL CALSTRN(DE,STRAIN)
C
      WRITE(LDEV2,11) CSTRN
C
C --- GET THE MATERIAL PARAMETERS
C
      CMO=MATL(1)
      CFO=MATL(2)
      EM=MATL(3)
      EF=MATL(4)
      UM=MATL(5)
      UF=MATL(6)
      GM=EM/(2.DO*(1.DO+UM))
      GF=EF/(2.DO*(1.DO+UF))
C
C --- GET THE YIELD PARAMETERS
C
      SY=MATL(7)
      B=MATL(8)
C
C --- GET THE DAMAGE PARAMETERS
C
      RM1=DMGPRM(1)
      RM2=DMGPRM(2)
      RM3=DMGPRM(3)
      QM1=DMGPRM(4)
      QM2=DMGPRM(5)
      QM3=DMGPRM(6)
      VM1=DMGPRM(7)
      VM2=DMGPRM(8)
      VM3=DMGPRM(9)
```

105

```
C
      RF1=DMGPRM(10)
      RF2=DMGPRM(11)
      RF3=DMGPRM(12)
      QF1=DMGPRM(13)
      QF2=DMGPRM(14)
      QF3=DMGPRM(15)
      VF1=DMGPRM(16)
      VF2=DMGPRM(17)
      VF3=DMGPRM(18)
C
      RB1=DMGPRM(19)
      RB2=DMGPRM(20)
      RB3=DMGPRM(21)
      QB1=DMGPRM(22)
      QB2=DMGPRM(23)
      QB3=DMGPRM(24)
      VB1=DMGPRM(25)
      VB2=DMGPRM(26)
      VB3=DMGPRM(27)
C ----------------------------------------------------------------------
C | CALCULATION OF THE STRESS INCREMENT & TOTAL STRESS ON EACH PLY     |
C ----------------------------------------------------------------------
C     TYPE OF LAMINATE
C       NTP=1 ; SINGLE LAMINAE
C       NTP=2 ; EVEN NUMBER OF LAMINAE
C       NTP=3 ; ODD NUMBER OF LAMINAE
C ----------------------------------------------------------------------
C
      DO 100 KP=1,NP
C
      IF (INCREM.GT.1) THEN
        READ(LDEV1,22) CAST,CYY,CPM,CPF,CPB,CSTM,CSTF,CCENT,CHKM,CHKF
     $  ,CHKB,CDMG,CDM
      ELSE
        DO 110 I=1,6
        AVGSTR(I)=0.DO
        PHIM(I)=0.DO
        PHIF(I)=0.DO
        PHIB(I)=0.DO
        STRSM(I)=0.DO
        STRSF(I)=0.DO
 110    CENT(I)=0.DO
        HKM=0.DO
        HKF=0.DO
        HKB=0.DO
        YY=0.DO
        CDMG=' '
        DO 115 I=1,6
        DO 115 J=1,6
        IF (I.EQ.J) THEN
          DM(I,J)=1.DO
        ELSE
          DM(I,J)=0.DO
        END IF
 115    CONTINUE
      END IF
C
      DO 120 I=1,6
 120  TSTRSM(I)=STRSM(I)
C
C --- GET THE FIBER DIRECTION OF PLY AND THICKNESS
C
      THETA=DEGRE(KP)
      THKK=PLYTHK(KP)
      SCALE=10.DO
C
C --- COMPUTE ELASTIC CONSTANTS
C
      CALL VOLFRC(CM,CF,PHIM,PHIF,CMO,CFO)
      CALL ADMAT(ESMB,EM,UM,GM)
      CALL ADMAT(ESFB,EF,UF,GF)
      CALL ASMAT(ECMB,EM,UM)
      CALL ASMAT(ECFB,EF,UF)
      IF (CDMG.NE.'Y') THEN
      CALL AECON(CMO,CFO,EM,GM,UM)
      CALL BECON(CMO,CFO)
      CALL PLYSTF(EPLYB,ESMB,ESFB,CMO,CFO)
      ELSE
      CALL AECON(CM,CF,EM,GM,UM)
```

106

```
        CALL BECON(CM,CF)
        CALL PLYSTF(EPLYB,ESMB,ESFB,CM,CF)
        END IF
        CALL COMPLI(ECB,EPLYB)
C
C --- COMPUTATION A TRIAL ELASTIC STRESS OF MATRIX
C
        IF (CDMG.NE.' ') THEN
          CALL DMAGE1(DMM,PHIM)
          CALL DMAGE1(DMF,PHIF)
          CALL DMAGE1(DMB,PHIB)
          CALL VOLFRC(CM,CF,PHIM,PHIF,CMO,CFO)
          CALL AECON(CM,CF,EM,GM,UM)
          CALL BECON(CM,CF)
          CALL ADCON(DM,DMM,DMF)
          CALL BDCON(DM,DMM,DMF)
          CALL ELDAMG(ESM,DMM,ESMB)
          CALL ELDAMG(ESF,DMF,ESFB)
          CALL PLYSTF(EPLY,ESM,ESF,CMO,CFO)
          CALL ELDAMG(DEPLY,DMB,EPLY)
          CALL TRNSTF(DEPLY,THETA)
          CALL CONVER(GRS,DEPLY)
        ELSE
          CALL AECON(CMO,CFO,EM,GM,UM)
          CALL BECON(CMO,CFO)
          CALL PLYSTF(EPLY,ESMB,ESFB,CMO,CFO)
          CALL TRNSTF(EPLY,THETA)
          CALL CONVER(GRS,EPLY)
        END IF
        CALL CALSTRS(DUMMY1,DUMMY2,DS,GRS,DE)
        CALL TRNSTR(TDS,DS,THETA)
        CALL LOSTRS(DSM,BME,TDS)
        CALL TRNSTN(TDE,DE,THETA)
        CALL LOSTRS(DSTRNM,AME,TDE)
C
        CALL UPDATE(TSTRSM,DSM)
C
C-----
        IF (CDMG.EQ.' ') THEN
C-----
        IF (YY.GT.0.9) THEN
          CALL AECON(CMO,CFO,EM,GM,UM)
          CALL BECON(CMO,CFO)
          CALL PLYSTF(EPLYB,ESMB,ESFB,CMO,CFO)
          CALL COMPLI(ECB,EPLYB)
          CALL ELPLD(PSMB,ESMB,STRSM,CENT,B)
          CALL APCON(PSMB,CMO,CFO,EM,EM,GM,GM,UM)
          CALL BPCON(PSMB,CMO,CFO)
          CALL PLYSTF(EPLY,PSMB,ESFB,CMO,CFO)
          CALL TRNSTF(EPLY,THETA)
          CALL CONVER(GRS,EPLY)
          CALL CALSTRS(DUMMY,AVGSTR,DS,GRS,DE)
          CALL TRNSTR(TDS,DS,THETA)
          CALL TRNSTR(TSTRS,AVGSTR,THETA)
          CALL LOSTRS(DSM,BME,TDS)
          CALL LOSTRS(DSF,BFE,TDS)
          CALL UPDATE(STRSM,DSM)
          CALL UPDATE(STRSF,DSF)
          CALL GRSSTF(DEP,GRS,NTP,KP,NP,THKK,THICK)
        ELSE
          CALL YIELD(FY,TSTRSM,CENT,SY)
          IF (FY.LT.0.0) THEN
            CALL AECON(CMO,CFO,EM,GM,UM)
            CALL BECON(CMO,CFO)
            CALL PLYSTF(EPLYB,ESMB,ESFB,CMO,CFO)
            CALL COMPLI(ECB,EPLYB)
            CALL PLYSTF(EPLY,ESMB,ESFB,CMO,CFO)
            CALL TRNSTF(EPLY,THETA)
            CALL CONVER(GRS,EPLY)
            CALL CALSTRS(DUMMY,AVGSTR,DS,GRS,DE)
            CALL TRNSTR(TDS,DS,THETA)
            CALL TRNSTR(TSTRS,AVGSTR,THETA)
            CALL LOSTRS(DSM,BME,TDS)
            CALL LOSTRS(DSF,BFE,TDS)
            CALL UPDATE(STRSM,DSM)
            CALL UPDATE(STRSF,DSF)
            CALL GRSSTF(DEP,GRS,NTP,KP,NP,THKK,THICK)
          ELSE
            YY=1.D0
            NSBI=INT(SCALE+0.1)
```

107

```
            DO 125 L=1,6
125         DDE(L)=DE(L)/SCALE
            CALL AECON(CMO,CFO,EM,GM,UM)
            CALL BECON(CMO,CFO)
            CALL PLYSTF(EPLYB,ESMB,ESFB,CMO,CFO)
            CALL COMPLI(ECB,EPLYB)
            CALL PLYSTF(EPLY,ESMB,ESFB,CMO,CFO)
            CALL TRNSTF(EPLY,THETA)
            CALL CONVER(GRS,EPLY)
            CALL CALSTRS(DUMMY1,AVGSTR,DDS,GRS,DDE)
            CALL TRNSTR(TDDS,DDS,THETA)
            CALL TRNSTR(TSTRS,AVGSTR,THETA)
            CALL LOSTRS(DDSM,BME,TDDS)
            CALL LOSTRS(DDSF,BFE,TDDS)
            CALL UPDATE(STRSM,DDSM)
            CALL UPDATE(STRSF,DDSF)
            DO 130 I=1,NSBI-1
            IF (IYY.EQ.10) THEN
              FY=1.0
            ELSE
              CALL YIELD(FY,STRSM,CENT,SY)
            END IF
            IF (FY.LT.0.0) THEN
            CALL AECON(CMO,CFO,EM,GM,UM)
            CALL BECON(CMO,CFO)
            CALL PLYSTF(EPLY,ESMB,ESFB,CMO,CFO)
            CALL TRNSTF(EPLY,THETA)
            CALL CONVER(GRS,EPLY)
            CALL CALSTRS(DUMMY,AVGSTR,DDS,GRS,DDE)
            CALL TRNSTR(TDDS,DDS,THETA)
            CALL TRNSTR(TSTRS,AVGSTR,THETA)
            CALL LOSTRS(DDSM,BME,TDDS)
            CALL LOSTRS(DDSF,BFE,TDDS)
            CALL UPDATE(STRSM,DDSM)
            CALL UPDATE(STRSF,DDSF)
            ELSE
            IYY=10
            CALL AECON(CMO,CFO,EM,GM,UM)
            CALL BECON(CMO,CFO)
            CALL ELPLD(PSMB,ESMB,STRSM,CENT,B)
            CALL APCON(PSMB,CMO,CFO,EM,EM,GM,GM,UM)
            CALL BPCON(PSMB,CMO,CFO)
            CALL PLYSTF(EPLY,PSMB,ESFB,CMO,CFO)
            CALL TRNSTF(EPLY,THETA)
            CALL CONVER(GRS,EPLY)
            CALL CALSTRS(DUMMY1,AVGSTR,DDS,GRS,DDE)
            CALL TRNSTR(TDDS,DDS,THETA)
            CALL TRNSTR(TSTRS,AVGSTR,THETA)
            CALL LOSTRS(DDSM,BME,TDDS)
            CALL LOSTRS(DDSF,BFE,TDDS)
            CALL UPDATE(STRSM,DDSM)
            CALL UPDATE(STRSF,DDSF)
            END IF
130         CONTINUE
            CALL GRSSTF(DEP,GRS,NTP,KP,NP,THKK,THICK)
         END IF
      END IF
C-----
      ELSE
C-----
C
      CALL DMAGE1(DMM,PHIM)
      CALL DMAGE1(DMF,PHIF)
      CALL DMAGE1(DMB,PHIB)
      CALL EFECT1(BSTRSM,DMM,STRSM)
      CALL EFECT1(BTSTRSM,DMM,TSTRSM)
      IF (YY.GT.0.9) THEN
        CALL VOLFRC(CM,CF,PHIM,PHIF,CMO,CFO)
        CALL AECON(CM,CF,EM,GM,UM)
        CALL BECON(CM,CF)
        CALL PLYSTF(EPLYB,ESMB,ESFB,CM,CF)
        CALL COMPLI(ECB,EPLYB)
        CALL ELPLD(PSMB,ESMB,BSTRSM,CENT,B)
        CALL APCON(PSMB,CM,CF,EM,EM,GM,GM,UM)
        CALL BPCON(PSMB,CM,CF)
        CALL ADCON(DM,DMM,DMF)
        CALL BDCON(DM,DMM,DMF)
        CALL ELDAMG(ESM,DMM,PSMB)
        CALL ELDAMG(ESF,DMF,ESFB)
        CALL PLYSTF(EPLY,ESM,ESF,CMO,CFO)
```

108

```
          CALL ELDAMG(DEPLY,DMB,EPLY)
          CALL TRNSTF(DEPLY,THETA)
          CALL CONVER(GRS,DEPLY)
          CALL CALSTRS(DUMMY,AVGSTR,DS,GRS,DE)
          CALL TRNSTR(TDS,DS,THETA)
          CALL TRNSTR(TSTRS,AVGSTR,THETA)
          CALL LOSTRS(DSM,BME,TDS)
          CALL LOSTRS(DSF,BFE,TDS)
          CALL UPDATE(STRSM,DSM)
          CALL UPDATE(STRSF,DSF)
          CALL GRSSTF(DEP,GRS,NTP,KP,NP,THKK,THICK)
        ELSE
          CALL YIELD(FY,BTSTRSM,CENT,SY)
          IF (FY.LT.0.0) THEN
            CALL VOLFRC(CM,CF,PHIM,PHIF,CMO,CFO)
            CALL AECON(CM,CF,EM,GM,UM)
            CALL BECON(CM,CF)
            CALL PLYSTF(EPLYB,ESMB,ESFB,CM,CF)
            CALL COMPLI(ECB,EPLYB)
            CALL ADCON(DM,DMM,DMF)
            CALL BDCON(DM,DMM,DMF)
            CALL ELDAMG(ESM,DMM,ESMB)
            CALL ELDAMG(ESF,DMF,ESFB)
            CALL PLYSTF(EPLY,ESM,ESF,CMO,CFO)
            CALL ELDAMG(DEPLY,DMB,EPLY)
            CALL TRNSTF(DEPLY,THETA)
            CALL CONVER(GRS,DEPLY)
            CALL CALSTRS(DUMMY,AVGSTR,DS,GRS,DE)
            CALL TRNSTR(TDS,DS,THETA)
            CALL TRNSTR(TSTRS,AVGSTR,THETA)
            CALL LOSTRS(DSM,BME,TDS)
            CALL LOSTRS(DSF,BFE,TDS)
            CALL UPDATE(STRSM,DSM)
            CALL UPDATE(STRSF,DSF)
            CALL GRSSTF(DEP,GRS,NTP,KP,NP,THKK,THICK)
          ELSE
            YY=1.D0
            NSBI=INT(SCALE+0.1)
            DO 135 L=1,6
135         DDE(L)=DE(L)/SCALE
            CALL VOLFRC(CM,CF,PHIM,PHIF,CMO,CFO)
            CALL AECON(CM,CF,EM,GM,UM)
            CALL BECON(CM,CF)
            CALL PLYSTF(EPLYB,ESMB,ESFB,CM,CF)
            CALL COMPLI(ECB,EPLYB)
            CALL ADCON(DM,DMM,DMF)
            CALL BDCON(DM,DMM,DMF)
            CALL ELDAMG(ESM,DMM,ESMB)
            CALL ELDAMG(ESF,DMF,ESFB)
            CALL PLYSTF(EPLY,ESM,ESF,CMO,CFO)
            CALL ELDAMG(DEPLY,DMB,EPLY)
            CALL TRNSTF(DEPLY,THETA)
            CALL CONVER(GRS,DEPLY)
            CALL CALSTRS(DUMMY1,AVGSTR,DDS,GRS,DDE)
            CALL TRNSTR(TDDS,DDS,THETA)
            CALL TRNSTR(TSTRS,AVGSTR,THETA)
            CALL LOSTRS(DDSM,BME,TDDS)
            CALL LOSTRS(DDSF,BFE,TDDS)
            CALL UPDATE(STRSM,DDSM)
            CALL UPDATE(STRSF,DDSF)
            DO 140 I=1,NSBI-1
            IF (IYY.EQ.10) THEN
              FY=1.0
            ELSE
              CALL EFECT1(BSTRSM,DMM,STRSM)
              CALL YIELD(FY,BSTRSM,CENT,SY)
            END IF
            IF (FY.LT.0.0) THEN
            CALL VOLFRC(CM,CF,PHIM,PHIF,CMO,CFO)
            CALL AECON(CM,CF,EM,GM,UM)
            CALL BECON(CM,CF)
            CALL ADCON(DM,DMM,DMF)
            CALL BDCON(DM,DMM,DMF)
            CALL ELDAMG(ESM,DMM,ESMB)
            CALL ELDAMG(ESF,DMF,ESFB)
            CALL PLYSTF(EPLY,ESM,ESF,CMO,CFO)
            CALL ELDAMG(DEPLY,DMB,EPLY)
            CALL TRNSTF(DEPLY,THETA)
            CALL CONVER(GRS,DEPLY)
            CALL CALSTRS(DUMMY,AVGSTR,DDS,GRS,DDE)
```

109

```
            CALL TRNSTR(TDDS,DDS,THETA)
            CALL TRNSTR(TSTRS,AVGSTR,THETA)
            CALL LOSTRS(DDSM,BME,TDDS)
            CALL LOSTRS(DDSF,BFE,TDDS)
            CALL UPDATE(STRSM,DDSM)
            CALL UPDATE(STRSF,DDSF)
            ELSE
            IYY=10
            CALL VOLFRC(CM,CF,PHIM,PHIF,CMO,CFO)
            CALL AECON(CM,CF,EM,GM,UM)
            CALL BECON(CM,CF)
            CALL ELPLD(PSMB,ESMB,BSTRSM,CENT,B)
            CALL APCON(PSMB,CM,CF,EM,EM,GM,GM,UM)
            CALL BPCON(PSMB,CM,CF)
            CALL ADCON(DM,DMM,DMF)
            CALL BDCON(DM,DMM,DMF)
            CALL ELDAMG(ESM,DMM,PSMB)
            CALL ELDAMG(ESF,DMF,ESFB)
            CALL PLYSTF(EPLY,ESM,ESF,CMO,CFO)
            CALL ELDAMG(DEPLY,DMB,EPLY)
            CALL TRNSTF(DEPLY,THETA)
            CALL CONVER(GRS,DEPLY)
            CALL CALSTRS(DUMMY1,AVGSTR,DDS,GRS,DDE)
            CALL TRNSTR(TDDS,DDS,THETA)
            CALL TRNSTR(TSTRS,AVGSTR,THETA)
            CALL LOSTRS(DDSM,BME,TDDS)
            CALL LOSTRS(DDSF,BFE,TDDS)
            CALL UPDATE(STRSM,DDSM)
            CALL UPDATE(STRSF,DDSF)
            END IF
  140       CONTINUE
            CALL GRSSTF(DEP,GRS,NTP,KP,NP,THKK,THICK)
         END IF
      END IF
C-----
      END IF
C-----
C
C -----------------------------------------------------------------------
C |                       UPDATE BACK-STRESS                             |
C -----------------------------------------------------------------------
C
      IF (YY.GT.0.9) THEN
      IF (CDMG.NE.'Y') THEN
        CALL FDER(FS,STRSM,CENT)
        CALL QSCALAR(Q,ESMB,STRSM,CENT,B)
        CALL LAMUDA(ALAM,ESMB,FS,DSTRNM,Q)
        CALL USCALAR(UT,STRSM,CENT,ALAM,B)
        CALL CENTER(CENT,STRSM,UT)
      ELSE
        CALL EFECT1(BSTRSM,DMM,STRSM)
        CALL EFECT2(BDSTRNM,DMM,DSTRNM)
        CALL FDER(FS,BSTRSM,CENT)
        CALL QSCALAR(Q,ESMB,BSTRSM,CENT,B)
        CALL LAMUDA(ALAM,ESMB,FS,BDSTRNM,Q)
        CALL USCALAR(UT,BSTRSM,CENT,ALAM,B)
        CALL CENTER(CENT,BSTRSM,UT)
      END IF
      END IF
C
C -----------------------------------------------------------------------
C |  UPDATE DMAGE ON EACH PLY                                            |
C -----------------------------------------------------------------------
C
      CALL DMAGE2(DMM,DMPM,PHIM)
      CALL DMAGE2(DMF,DMPF,PHIF)
      CALL DMAGE2(DMB,DMPB,PHIB)
C
      CALL DMGFRC(YM,DMM,DMPM,STRSM,ECMB)
      CALL DMGFRC(YF,DMF,DMPF,STRSF,ECFB)
      CALL DMGFRC(YB,DMB,DMPB,TSTRS,ECB)
C
      U1=ESMB(1,1)-2.D0*ESMB(4,4)
      U2=ESMB(2,2)-2.D0*ESMB(5,5)
      U3=ESMB(3,3)-2.D0*ESMB(6,6)
C
      CALL DMGHAR(HMI,AM1,AM2,AM3,BM1,BM2,BM3,PHIM,HKM,RM1,RM2,RM3
     $,QM1,QM2,QM3,U1,U2,U3,VM1,VM2,VM3)
C
      U1=ESFB(1,1)-2.D0*ESFB(4,4)
```

```
      U2=ESFB(2,2)-2.D0*ESFB(5,5)
      U3=ESFB(3,3)-2.D0*ESFB(6,6)
C
      CALL DMGHAR(HFI,AF1,AF2,AF3,BF1,BF2,BF3,PHIF,HKF,RF1,RF2,RF3
     $,QF1,QF2,QF3,U1,U2,U3,VF1,VF2,VF3)
C
      U1=EPLYB(1,1)-2.D0*EPLYB(4,4)
      U2=EPLYB(2,2)-2.D0*EPLYB(5,5)
      U3=EPLYB(3,3)-2.D0*EPLYB(6,6)
C
      CALL DMGHAR(HBI,AB1,AB2,AB3,BB1,BB2,BB3,PHIB,HKB,RB1,RB2,RB3
     $,QB1,QB2,QB3,U1,U2,U3,VB1,VB2,VB3)
C
      CALL DMGCRT(GCRTM,YM,HMI)
      CALL DMGCRT(GCRTF,YF,HFI)
      CALL DMGCRT(GCRTB,YB,HBI)
C
      IF (GCRTM.GE.1.0) THEN
        CALL CALDMG(PHIM,DMM,DMPM,STRSM,DSM,ECMB,HKM,YM,HMI
     $  ,AM1,AM2,AM3,BM1,BM2,BM3,CDMG,1)
        CDMG='Y'
      END IF
C
      IF (GCRTF.GE.1.0) THEN
        CALL CALDMG(PHIF,DMF,DMPF,STRSF,DSF,ECFB,HKF,YF,HFI
     $  ,AF1,AF2,AF3,BF1,BF2,BF3,CDMG,2)
        CDMG='Y'
      END IF
C
      IF (GCRTB.GE.1.0) THEN
        CALL CALDMG(PHIB,DMB,DMPB,TSTRS,TDS,ECB,HKB,YB,HBI
     $  ,AB1,AB2,AB3,BB1,BB2,BB3,CDMG,3)
        CDMG='Y'
      END IF
C
      IF (CDMG.NE.' ') THEN
        CALL DMAGE1(DMM,PHIM)
        CALL DMAGE1(DMF,PHIF)
        CALL VOLFRC(CM,CF,PHIM,PHIF,CMO,CFO)
        CALL OVDMG(DM,DMM,DMF,BME,BFE,CM,CF)
        CALL AECON(CM,CF,EM,GM,UM)
        CALL BECON(CM,CF)
        CALL BDCON(DM,DMM,DMF)
        CALL OVDMG(DM,DMM,DMF,BME,BFE,CM,CF)
      END IF
C
      WRITE(LDEV2,22) CAST,CYY,CPM,CPF,CPB,CSTM,CSTF,CCENT,CHKM,CHKF
     $,CHKB,CDMG,CDM
C
 100  CONTINUE
C
C -----------------------------------------------------------------------
C |   UPDATE AVERAGE STRESSES  OF LAMINATE                                |
C -----------------------------------------------------------------------
C
      IF (INCREM.GT.1) THEN
        READ(LDEV1,11) CSTRS
      ELSE
        DO 200 K1=1,6
 200    STRESS(K1)=0.D0
      END IF
C
      CALL CALSTRS(STRS,STRESS,DUMMY,DEP,DE)
C
      WRITE(LDEV2,11) CSTRS
C
      CALL CHKCRK(NP,NCRACK,ICRK)
C
      IF (ICRK.EQ.'Y') THEN
        WRITE(6,878) ELNUM,INTGPN
 878    FORMAT(2X,'LAMINATE CRACK',2X,I5,2X,I2)
      END IF
C
      RETURN
C
C
C ======================================================================
C ===================== ENTRY LCALSTF ==================================
C ======================================================================
      ENTRY LCALSTF(ELNUM,INTGPN)
```

111

```
C
      IF (INCREM.GT.1) THEN
        IF (NIT.EQ.1) THEN
          READ(LDEV1,11) CSTRN
        ELSE
          READ(LDEV2,11) CSTRN
        END IF
      END IF
C
C --- GET THE MATERIAL PARAMETERS
C
      CMO=0.65DO
      CFO=0.35DO
      EM=80000.DO
      EF=410000.DO
      UM=0.3DO
      UF=0.22DO
      GM=EM/(2.DO*(1.DO+UM))
      GF=EF/(2.DO*(1.DO+UF))
C
C --- GET THE YIELD PARAMETERS
C
      B=90.DO
C
C --- COMPUTE ELASTIC CONSTANTS
C
      CALL ADMAT(ESMB,EM,UM,GM)
      CALL ADMAT(ESFB,EF,UF,GF)
C
C ------------------------------------------------------------------
C    TYPE OF LAMINATE
C      NTP=1 ; SINGLE LAMINAE
C      NTP=2 ; EVEN NUMBER OF LAMINAE
C      NTP=3 ; ODD NUMBER OF LAMINAE
C ------------------------------------------------------------------
C
      DO 300 LP=1,NP
C
      IF (INCREM.GT.1) THEN
      IF (NIT.EQ.1) THEN
       READ(LDEV1,22) CAST,CYY,CPM,CPF,CPB,CSTM,CSTF,CCENT,CHKM,CHKF
     $,CHKB,CDMG,CDM
      ELSE
       READ(LDEV2,22) CAST,CYY,CPM,CPF,CPB,CSTM,CSTF,CCENT,CHKM,CHKF
     $,CHKB,CDMG,CDM
      END IF
      ELSE
       CDMG=' '
       YY=0.DO
      END IF
C
C --- GET THE FIBER DIRECTION OF PLY AND THICKNESS
C
      THETA=DEGRE(LP)
      THKK=PLYTHK(LP)
C
C-----
      IF (CDMG.NE.'Y') THEN
C-----
      IF (YY.GT.0.9) THEN
        CALL AECON(CMO,CFO,EM,GM,UM)
        CALL ELPLD(PSMB,ESMB,STRSM,CENT,B)
        CALL APCON(PSMB,CMO,CFO,EM,EM,GM,GM,UM)
        CALL PLYSTF(EPLY,PSMB,ESFB,CMO,CFO)
        CALL TRNSTF(EPLY,THETA)
        CALL CONVER(GRS,EPLY)
        CALL GRSSTF(DEP,GRS,NTP,LP,NP,THKK,THICK)
      ELSE
        CALL AECON(CMO,CFO,EM,GM,UM)
        CALL PLYSTF(EPLY,ESMB,ESFB,CMO,CFO)
        CALL TRNSTF(EPLY,THETA)
        CALL CONVER(GRS,EPLY)
        CALL GRSSTF(DEP,GRS,NTP,LP,NP,THKK,THICK)
C
      END IF
C-----
      ELSE
C-----
C
      CALL DMAGE1(DMM,PHIM)
```

112

```fortran
          CALL DMAGE1(DMF,PHIF)
          CALL DMAGE1(DMB,PHIB)
          CALL EFECT1(BSTRSM,DMM,STRSM)
          IF (YY.GT.0.9) THEN
            CALL VOLFRC(CM,CF,PHIM,PHIF,CMO,CFO)
            CALL AECON(CM,CF,EM,GM,UM)
            CALL ELPLD(PSMB,ESMB,BSTRSM,CENT,B)
            CALL APCON(PSMB,CM,CF,EM,EM,GM,GM,UM)
            CALL ADCON(DM,DMM,DMF)
            CALL ELDAMG(ESM,DMM,PSMB)
            CALL ELDAMG(ESF,DMF,ESFB)
            CALL PLYSTF(EPLY,ESM,ESF,CMO,CFO)
            CALL ELDAMG(DEPLY,DMB,EPLY)
            CALL TRNSTF(DEPLY,THETA)
            CALL CONVER(GRS,DEPLY)
            CALL GRSSTF(DEP,GRS,NTP,LP,NP,THKK,THICK)
          ELSE
            CALL VOLFRC(CM,CF,PHIM,PHIF,CMO,CFO)
            CALL AECON(CM,CF,EM,GM,UM)
            CALL ADCON(DM,DMM,DMF)
            CALL ELDAMG(ESM,DMM,ESMB)
            CALL ELDAMG(ESF,DMF,ESFB)
            CALL PLYSTF(EPLY,ESM,ESF,CMO,CFO)
            CALL ELDAMG(DEPLY,DMB,EPLY)
            CALL TRNSTF(DEPLY,THETA)
            CALL CONVER(GRS,DEPLY)
            CALL GRSSTF(DEP,GRS,NTP,LP,NP,THKK,THICK)
          END IF
C-----
          END IF
C-----
C
 300    CONTINUE
C
        IF (INCREM.GT.1) THEN
          IF (NIT.EQ.1) THEN
            READ(LDEV1,11) CSTRS
          ELSE
            READ(LDEV2,11) CSTRS
          END IF
          BACKSPACE(UNIT=LDEV)
          BACKSPACE(UNIT=LDEV)
          BACKSPACE(UNIT=LDEV)
          BACKSPACE(UNIT=LDEV)
        END IF
C
        RETURN
C
 11     FORMAT(A48)
 22     FORMAT(A48,A8,6A48,3A8,A1,A288)
        END
C =======================================================================
C ====================== C A L S T R N ==================================
C =======================================================================
        SUBROUTINE CALSTRN(DE,STRAIN)
C
C --- CALCULATION OF THE STRAIN INCREMENT
C
        IMPLICIT REAL*8 (A-H,O-Z)
        COMMON/ELSTR1/STRN(6)
        DIMENSION DE(6),STRAIN(6)
        DO 10 K1=1,3
        DE(K1)=STRN(K1)-STRAIN(K1)
 10     STRAIN(K1)=STRN(K1)
        RETURN
        END
C =======================================================================
C ====================== C A L S T R S ==================================
C =======================================================================
        SUBROUTINE CALSTRS(STRS,STRESS,DS,DEP,DE)
C THIS SUBPROGRAM CALCULATES THE STRESS INCREMENT AND
C THE TOTAL STRESS
        IMPLICIT REAL*8 (A-H,O-Z)
        DIMENSION STRS(6),STRESS(6),DS(6),DEP(6,6),DE(6)
C
        DO 10 K1=1,3
        CST=0.DO
        DO 15 K2=1,3
 15     CST=CST+DEP(K1,K2)*DE(K2)
 10     DS(K1)=CST
```

113

```
C
      DO 20 K1=1,3
      STRESS(K1)=STRESS(K1)+DS(K1)
  20  STRS(K1)=STRESS(K1)
C
      RETURN
      END
C ================================================================
C ==================== SUBROUTINE UPDATE  ========================
C ================================================================
      SUBROUTINE UPDATE(A,B)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION A(6),B(6)
      DO 10 I=1,6
  10  A(I)=A(I)+B(I)
      RETURN
      END
C ================================================================
C ==================== SUBROUTINE CHKCRK =========================
C ================================================================
      SUBROUTINE CHKCRK(NP,NCRACK,ICRK)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION NCRACK(5)
      CHARACTER*1 ICRK
      IF((NCRACK(1).EQ.1).AND.(NCRACK(2).EQ.1)) THEN
        ICRK='Y'
      ELSE
        ICRK=' '
      END IF
      RETURN
      END
C ================================================================
C ====================== A D M A T ===============================
C ================================================================
      SUBROUTINE ADMAT(AD,YOUNG,POISS,AMUE)
C ================================================================
C I                                                             I
C I    P R O G R A M:                                           I
C I    'ADMAT' CALCULATES THE ELASTIC STIFFNESS MATRIX.         I
C I    A R G U M E N T    L I S T:                              I
C I    YOUNG = YOUGS MODULUS                                    I
C I    POISS = POISSONS RATIO                                   I
C ================================================================
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION AD(6,6)
C
C --- ALAM =   THE LAMDA LAME CONSTANT
C --- AMUE =   THE MU LAME CONSTANT (THE SHEAR MODULUS G)
C
      ALAM=POISS*YOUNG/(1.DO+POISS)/(1.DO-2.DO*POISS)
C
      AD(1,1)=ALAM+2.DO*AMUE
      AD(1,2)=ALAM
      AD(1,3)=ALAM
      AD(2,1)=ALAM
      AD(2,2)=ALAM+2.DO*AMUE
      AD(2,3)=ALAM
      AD(3,1)=ALAM
      AD(3,2)=ALAM
      AD(3,3)=ALAM+2.DO*AMUE
      AD(4,4)=AMUE
      AD(5,5)=AMUE
      AD(6,6)=AMUE
C
      RETURN
      END
C ================================================================
C ====================== A S M A T ===============================
C ================================================================
      SUBROUTINE ASMAT(AS,YOUNG,POISS)
C ================================================================
C I    P R O G R A M:                                           I
C I    'ADMAT' CALCULATES THE ELASTIC COMPLIANCE MATRIX.        I
C I    A R G U M E N T    L I S T:                              I
C I    YOUNG = YOUGS MODULUS                                    I
C I    POISS = POISSONS RATIO                                   I
C ================================================================
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION AS(6,6)
      AS(1,1)=1.DO/YOUNG
```

114

```
      AS(1,2)=-POISS/YOUNG
      AS(1,3)=AS(1,2)
      AS(2,1)=AS(1,2)
      AS(2,2)=AS(1,1)
      AS(2,3)=AS(1,2)
      AS(3,1)=AS(1,2)
      AS(3,2)=AS(1,2)
      AS(3,3)=AS(1,1)
      AS(4,4)=(2.DO*(1.DO+POISS))/YOUNG
      AS(5,5)=AS(4,4)
      AS(6,6)=AS(4,4)
      RETURN
      END
C ======================================================================
C ========================= C O M P L I ================================
C ======================================================================
      SUBROUTINE COMPLI(ECB,EPLY)
C ======================================================================
C I   P R O G R A M:                                                  I
C I   'COMPLI CALCULATES THE ELASTIC COMPLIANCE MATRIX OF COMPOSITE   I
C ======================================================================
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION ECB(6,6),EPLY(6,6)
      DO 10 I=1,6
      DO 10 J=1,6
  10  ECB(I,J)=EPLY(I,J)
      CALL AINV(ECB)
      RETURN
      END
C ======================================================================
C ======================= L O S T R N ==================================
C ======================================================================
      SUBROUTINE LOSTRN(TLE,AE,TDE)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION TLE(6),AE(6,6),TDE(6)
      CALL MULV1(TLE,AE,TDE)
      RETURN
      END
C ======================================================================
C ======================= L O S T R S ==================================
C ======================================================================
      SUBROUTINE LOSTRS(S,BE,STRESS)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION S(6),BE(6,6),STRESS(6)
      CALL MULV1(S,BE,STRESS)
      RETURN
      END
C ======================================================================
C ======================= E L D A M G ==================================
C ======================================================================
      SUBROUTINE ELDAMG(ED,DM,E)
C
C THIS SUBPROGRAM CALCULATES THE DAMAGED ELASTIC STIFFNESS FOR
C MATRIX OR FIBERS
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION ED(6,6),DM(6,6),E(6,6)
      DIMENSION DMI(6,6),DMIT(6,6),DMIE(6,6)
C
      CALL DMINV(DMI,DM)
      CALL TRANR(DMIT,DMI)
      CALL ATIMB(DMIE,DMI,E)
      CALL ATIMB(ED,DMIE,DMIT)
C
      RETURN
      END
C ======================================================================
C ======================= P L Y S T F  =================================
C ======================================================================
      SUBROUTINE PLYSTF(EPLY,ESM,ESF,CM,CF)
C
C THIS SUBPROGRAM CALCULATES THE LAMINAE ELASTIC STIFFNESS
C MATRIX IN THE MATERIAL DIRECTION
C
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/AMECON1/AME(6,6)
      COMMON/AFECON1/AFE(6,6)
      DIMENSION ADEM(6,6),ADEF(6,6),EPLY(6,6),ESM(6,6),ESF(6,6)
C
      CALL ATIMB(ADEM,ESM,AME)
```

115

```
      CALL ATIMB(ADEF,ESF,AFE)
C
      DO 10 I=1,6
      DO 10 J=1,6
   10 EPLY(I,J)=CM*ADEM(I,J)+CF*ADEF(I,J)
C
      RETURN
      END
C
C ==========================================================================
C ====================== G R S S T F ===============================
C ==========================================================================
      SUBROUTINE GRSSTF(GRS,Q,NTP,KP,NP,THKK,THICK)
C
C THIS SUBPROGRAM CALCULATES THE GROSS ELASTIC STIFFNESS
C MATRIX INCLUDING THE EFFECT OF DAMAGE FOR LAMINATE
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION GRS(6,6),Q(6,6),A(6,6)
C
      IF (NTP.EQ.1) THEN
        DO 10 I=1,3
        DO 10 J=1,3
        GRS(I,J)=Q(I,J)
   10   CONTINUE
        GO TO 100
      END IF
C
      IF (KP.EQ.1) THEN
        DO 20 I=1,3
        DO 20 J=1,3
        A(I,J)=0.DO
   20   CONTINUE
      END IF
C
      DO 30 I=1,3
      DO 30 J=1,3
      IF ((KP.EQ.NP).AND.(NTP.EQ.3)) THEN
        A(I,J)=A(I,J)+Q(I,J)*THKK/2.DO
      ELSE
        A(I,J)=A(I,J)+Q(I,J)*THKK
      END IF
   30 CONTINUE
C
      IF (KP.EQ.NP) THEN
        DO 40 I=1,3
        DO 40 J=1,3
        GRS(I,J)=2.DO*A(I,J)/THICK
   40   CONTINUE
      END IF
C
  100 RETURN
      END
C ==========================================================================
C ====================== T R N S T R ==============================
C ==========================================================================
      SUBROUTINE TRNSTR(TE,E,THETA)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION TE(6),E(6)
C
      PI=4.0DO*DATAN(1.0DO)
      RADIAN=THETA*PI/180.0DO
      C=DCOS(2.DO*RADIAN)
      S=DSIN(2.DO*RADIAN)
C
      EADD=0.5DO*(E(1)+E(2))
      ESUB=0.5DO*(E(1)-E(2))
C
      TE(1)=EADD+ESUB*C+E(3)*S
      TE(2)=EADD-ESUB*C-E(3)*S
      TE(4)=-ESUB*S+E(3)*C
      TE(3)=0.DO
      TE(5)=0.DO
      TE(6)=0.DO
C
      RETURN
      END
C
C ==========================================================================
C ====================== T R N S T N ==============================
```

```fortran
C ======================================================================
      SUBROUTINE TRNSTN(TE,E,THETA)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION TE(6),E(6)
C
      PI=4.0D0*DATAN(1.0D0)
      RADIAN=THETA*PI/180.0D0
      C=DCOS(2.D0*RADIAN)
      S=DSIN(2.D0*RADIAN)
C
      T=0.5D0*E(3)
C
      EADD=0.5D0*(E(1)+E(2))
      ESUB=0.5D0*(E(1)-E(2))
C
      TE(1)=EADD+ESUB*C+T*S
      TE(2)=EADD-ESUB*C-T*S
      TE(4)=-ESUB*S+T*C
      TE(4)=TE(4)*2.D0
      TE(3)=0.D0
      TE(5)=0.D0
      TE(6)=0.D0
      RETURN
      END
C ======================================================================
C ========================= Y I E L D ==================================
C ======================================================================
      SUBROUTINE YIELD(FY,S,ALPHA,SY)
C ======================================================================
C I                                                                    I
C I    THIS SUBPROGRAM CALCULATES THE VALUE OF THE YIELD FUNCTION.     I
C I    THE PROGRAMED YIED FUNCTION IS AN EXTENDE FORM OF THE           I
C I    VON MISES YIELD CRITERION. THIS YIELD FUNCTION IS THE           I
C I    EQUIVALANT LAGRANGIAN FORMULATION OF THE EULERIAN VON MISES     I
C I    TYPE YIELD CRITERIA.                                            I
C I                                                                    I
C I    THE YIELD FUNCTION HAS THE FOLLOWING FORM.                      I
C I                                                                    I
C I    SY        IS THE YIELD STRESS IN SIMPLE TENSION TEST            I
C I                                                                    I
C ======================================================================
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION S(6),ALPHA(6),TAU(6)
      DO 10 I=1,6
   10 TAU(I)=S(I)-ALPHA(I)
      CALL SCLVV(FY,TAU,TAU)
      FY=3.D0*FY/2.D0-SY*SY
      RETURN
      END
C ======================================================================
C ========================= F D E R ====================================
C ======================================================================
      SUBROUTINE FDER(FS,S,ALPHA)
C ======================================================================
C I      THIS PROGRAM CALCULATES THE DERIVATIVE OF "F" WRT            I
C I      <STRESS>                                                     I
C I      FS = DERIVATIVE OF F WRT <STRESS>                            I
C ======================================================================
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION FS(6),S(6),ALPHA(6)
      DO 10 I=1,6
   10 FS(I)=3.D0*(S(I)-ALPHA(I))
      RETURN
      END
C ======================================================================
C ========================= E L P L D ==================================
C ======================================================================
      SUBROUTINE ELPLD(EP,E,S,ALPHA,B)
C ======================================================================
C I                                                                    I
C I      THIS PROGRAM CALCULATES THE ELASTOPLASTIC MATRIX             I
C I      THAT CORRESPONDS TO THE YIELD FUNCTION F                     I
C I                                                                    I
C ======================================================================
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION EP(6,6),E(6,6),FS(6),S(6),ALPHA(6)
      DIMENSION TEMP1(6),TEMP2(6),P(6,6)
C
      DO 10 I=1,6
   10 FS(I)=3.D0*(S(I)-ALPHA(I))
```

117

```
C
      CALL QSCALAR(Q,E,S,ALPHA,B)
      CALL MULV1(TEMP1,E,FS)
      CALL MULVV(P,TEMP1,TEMP1)
C
      DO 20 I=1,6
      DO 20 J=1,6
  20  P(I,J)=P(I,J)/Q
C
      CALL SUBMT(EP,E,P)
C
      RETURN
      END
C
C =====================================================================
C ======================= Q S C A L A R =============================
C =====================================================================
      SUBROUTINE QSCALAR(Q,E,S,ALPHA,B)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION FS(6),FA(6),E(6,6),S(6),ALPHA(6),TAU(6),TEMP(6)
      DO 10 I=1,6
      FS(I)=3.DO*(S(I)-ALPHA(I))
      FA(I)=-FS(I)
  10  TAU(I)=S(I)-ALPHA(I)
      CALL MULV2(TEMP,FS,E)
      CALL SCLVV(Q1,TEMP,FS)
      CALL SCLVV(Q2,FA,TAU)
      CALL SCLVV(Q3,FS,FS)
      CALL SCLVV(Q4,TAU,FS)
C
      Q=Q1-Q2*B*Q3/Q4
C
      RETURN
      END
C =====================================================================
C ======================= U S C A L A R =============================
C =====================================================================
      SUBROUTINE USCALAR(UT,S,ALPHA,ALAM,B)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION FS(6),S(6),ALPHA(6),TAU(6)
      DO 10 I=1,6
      FS(I)=3.DO*(S(I)-ALPHA(I))
  10  TAU(I)=S(I)-ALPHA(I)
      CALL SCLVV(UT1,FS,FS)
      CALL SCLVV(UT2,TAU,FS)
      UT=ALAM*B*UT1/UT2
      RETURN
      END
C =====================================================================
C ======================= L A M U D A ===============================
C =====================================================================
      SUBROUTINE LAMUDA(ALAM,E,FS,DE,Q)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION E(6,6),FS(6),DE(6),TEMP(6)
      CALL MULV2(TEMP,FS,E)
      CALL SCLVV(ALAM,TEMP,DE)
      ALAM=ALAM/Q
      RETURN
      END
C =====================================================================
C ======================= C E N T E R ===============================
C =====================================================================
      SUBROUTINE CENTER(ALPHA,S,UT)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION ALPHA(6),DALPHA(6),S(6),TEMP(6)
      DO 10 I=1,6
  10  TEMP(I)=S(I)-ALPHA(I)
      DO 20 I=1,6
  20  DALPHA(I)=TEMP(I)*UT
      DO 30 I=1,6
  30  ALPHA(I)=ALPHA(I)+DALPHA(I)
      RETURN
      END
C =====================================================================
C ======================= E L D A M G ===============================
C =====================================================================
      SUBROUTINE ELDAMG(ED,DM,E)
C
C THIS SUBPROGRAM CALCULATES THE DAMAGED ELASTIC STIFFNESS FOR
C MATRIX OR FIBERS
```

118

```
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION ED(6,6),DM(6,6),E(6,6)
      DIMENSION DMI(6,6),DMIT(6,6),DMIE(6,6)
      CALL DMINV(DMI,DM)
      CALL TRANR(DMIT,DMI)
      CALL ATIMB(DMIE,DMI,E)
      CALL ATIMB(ED,DMIE,DMIT)
      RETURN
      END
C ================================================================
C ===================== E I D A M G ==============================
C ================================================================
      SUBROUTINE EIDAMG(EID,DM,EIB)
C
C THIS SUBPROGRAM CALCULATES THE DAMAGED ELASTIC COMPLIANCE
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION EID(6,6),DM(6,6),EIB(6,6)
      DIMENSION DMT(6,6),DMTE(6,6)
      CALL TRANR(DMT,DM)
      CALL ATIMB(DMTE,DMT,EIB)
      CALL ATIMB(EID,DMTE,DM)
      RETURN
      END
C ================================================================
C ===================== A E C O N ================================
C ================================================================
      SUBROUTINE AECON(CM,CF,EA,GA,UA)
      IMPLICIT REAL*8 (A-H,O-Z)
      CALL CKLMP(EA,GA,UA,PK,PM,PP)
      CALL PCONCEN(PK,PM,PP)
      CALL TFCON
      CALL AMECON(CM,CF)
      CALL AFECON
      RETURN
      END
C ================================================================
C ===================== C K L M P ================================
C ================================================================
      SUBROUTINE CKLMP(EA,GA,UA,XK,XM,XP)
      IMPLICIT REAL*8 (A-H,O-Z)
      XK=EA/(3.D0*(1.D0-2.D0*UA))+GA/3.D0
      XM=EA/(2.D0*(1.D0+UA))
      XP=XM
      RETURN
      END
C ================================================================
C ===================== P C O N C E N ============================
C ================================================================
      SUBROUTINE PCONCEN(PK,PM,PP)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/PCON1/P(6,6)
      P(2,2)=(PK+4.D0*PM)/(8.D0*PM*(PK+PM))
      P(3,3)=P(2,2)
      P(2,3)=-PK/(8.D0*PM*(PK+PM))
      P(3,2)=P(2,3)
      P(4,4)=(PK+2.D0*PM)/(2.D0*PM*(PK+PM))
      P(5,5)=1.D0/(2.D0*PP)
      P(6,6)=P(5,5)
      RETURN
      END
C ================================================================
C ===================== A F E C O N ==============================
C ================================================================
      SUBROUTINE AFECON
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/AFECON1/AFE(6,6)
      COMMON/AMECON1/AME(6,6)
      COMMON/TFCON1/TF(6,6)
C
      CALL ATIMB(AFE,TF,AME)
C
      RETURN
      END
C
C ================================================================
C ===================== A M E C O N ==============================
C ================================================================
      SUBROUTINE AMECON(CM,CF)
```

```fortran
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/AMECON1/AME(6,6)
      COMMON/TFCON1/TF(6,6)
C
      DO 10 I=1,6
      DO 10 J=1,6
   10 AME(I,J)=CF*TF(I,J)
C
      AME(1,1)=CM+AME(1,1)
      AME(2,2)=CM+AME(2,2)
      AME(3,3)=CM+AME(3,3)
      AME(4,4)=CM+AME(4,4)
      AME(5,5)=CM+AME(5,5)
      AME(6,6)=CM+AME(6,6)
C
      CALL AIINV(AME)
C
      RETURN
      END
C
C =====================================================================
C ======================= T F C O N ===================================
C =====================================================================
      SUBROUTINE TFCON
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/MEDAM1/ESMB(6,6),ESFB(6,6)
      COMMON/TFCON1/TF(6,6)
      COMMON/PCON1/P(6,6)
C     COMMON/DFES1/DFES(6,6)
      DIMENSION DFES(6,6)
C
      DFES(1,1)=ESFB(1,1)-ESMB(1,1)
      DFES(1,2)=ESFB(1,2)-ESMB(1,2)
      DFES(1,3)=ESFB(1,3)-ESMB(1,3)
      DFES(2,1)=DFES(1,2)
      DFES(3,1)=DFES(1,3)
      DFES(2,2)=ESFB(2,2)-ESMB(2,2)
      DFES(2,3)=ESFB(2,3)-ESMB(2,3)
      DFES(3,2)=DFES(2,3)
      DFES(3,3)=ESFB(3,3)-ESMB(3,3)
      DFES(4,4)=ESFB(4,4)-ESMB(4,4)
      DFES(5,5)=ESFB(5,5)-ESMB(5,5)
      DFES(6,6)=ESFB(6,6)-ESMB(6,6)
C
      CALL ATIMB(TF,P,DFES)
C
      TF(1,1)=1.D0+TF(1,1)
      TF(2,2)=1.D0+TF(2,2)
      TF(3,3)=1.D0+TF(3,3)
      TF(4,4)=1.D0+TF(4,4)
      TF(5,5)=1.D0+TF(5,5)
      TF(6,6)=1.D0+TF(6,6)
C
      CALL AIINV(TF)
C
      RETURN
      END
C =====================================================================
C ======================= A P C O N ===================================
C =====================================================================
      SUBROUTINE APCON(PSMB,CM,CF,EA,ET,GA,GT,UA)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION PSMB(6,6)
C
      CALL CKLMP(EA,ET,GA,GT,UA,PK,PL,PM,PP,PN)
      CALL PCONCEN(PK,PM,PP,PN)
      CALL PTFCON(PSMB)
      CALL AMECON(CM,CF)
      CALL AFECON
C
      RETURN
      END
C =====================================================================
C ======================= P T F C O N =================================
C =====================================================================
      SUBROUTINE PTFCON(PSMB)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/MEDAM1/ESMB(6,6),ESFB(6,6)
      COMMON/TFCON1/TF(6,6)
      COMMON/PCON1/P(6,6)
```

```
C       COMMON/DFES1/DFES(6,6)
        DIMENSION PSMB(6,6),DFES(6,6)
C
        DO 10 I=1,6
        DO 10 J=1,6
  10    DFES(I,J)=ESFB(I,J)-PSMB(I,J)
C
        CALL ATIMB(TF,P,DFES)
C
        TF(1,1)=1.D0+TF(1,1)
        TF(2,2)=1.D0+TF(2,2)
        TF(3,3)=1.D0+TF(3,3)
        TF(4,4)=1.D0+TF(4,4)
        TF(5,5)=1.D0+TF(5,5)
        TF(6,6)=1.D0+TF(6,6)
C
        CALL AIINV(TF)
C
        RETURN
        END
C
C ========================================================================
C ========================= Q C O N C E N ===============================
C ========================================================================
        SUBROUTINE QCONCEN
        IMPLICIT REAL*8 (A-H,O-Z)
        COMMON/QCON1/Q(6,6)
        COMMON/PCON1/P(6,6)
        COMMON/MEDAM1/ESMB(6,6),ESFB(6,6)
        DIMENSION EP(6,6),Q1(6,6)
C
        CALL ATIMB(EP,ESMB,P)
        CALL ATIMB(Q1,EP,ESMB)
        CALL SUBMT(Q,ESMB,Q1)
C
        RETURN
        END
C
C ========================================================================
C ========================= W F C O N ===================================
C ========================================================================
        SUBROUTINE WFCON
        IMPLICIT REAL*8 (A-H,O-Z)
        COMMON/MEDAM2/ECMB(6,6),ECFB(6,6)
        COMMON/WFCON1/WF(6,6)
        COMMON/QCON1/Q(6,6)
C       COMMON/DFEC1/DFEC(6,6)
        DIMENSION DFEC(6,6)
C
        DFEC(1,1)=ECFB(1,1)-ECMB(1,1)
        DFEC(1,2)=ECFB(1,2)-ECMB(1,2)
        DFEC(1,3)=ECFB(1,3)-ECMB(1,3)
        DFEC(2,1)=DFEC(1,2)
        DFEC(3,1)=DFEC(1,3)
        DFEC(2,2)=ECFB(2,2)-ECMB(2,2)
        DFEC(2,3)=ECFB(2,3)-ECMB(2,3)
        DFEC(3,2)=DFEC(2,3)
        DFEC(3,3)=ECFB(3,3)-ECMB(3,3)
        DFEC(4,4)=ECFB(4,4)-ECMB(4,4)
        DFEC(5,5)=ECFB(5,5)-ECMB(5,5)
        DFEC(6,6)=ECFB(6,6)-ECMB(6,6)
C
        CALL ATIMB(WF,Q,DFEC)
C
        WF(1,1)=1.D0+WF(1,1)
        WF(2,2)=1.D0+WF(2,2)
        WF(3,3)=1.D0+WF(3,3)
        WF(4,4)=1.D0+WF(4,4)
        WF(5,5)=1.D0+WF(5,5)
        WF(6,6)=1.D0+WF(6,6)
C
        CALL AIINV(WF)
C
        RETURN
        END
C ========================================================================
C ========================= B F E C O N =================================
C ========================================================================
        SUBROUTINE BFECON
        IMPLICIT REAL*8 (A-H,O-Z)
```

121

```
          COMMON/BFECON1/BFE(6,6)
          COMMON/BMECON1/BME(6,6)
          COMMON/WFCON1/WF(6,6)
C
          CALL ATIMB(BFE,WF,BME)
C
          RETURN
          END
C
C ======================================================================
C ======================== B M E C O N  ============================
C ======================================================================
          SUBROUTINE BMECON(CM,CF)
          IMPLICIT REAL*8 (A-H,O-Z)
          COMMON/BMECON1/BME(6,6)
          COMMON/WFCON1/WF(6,6)
C
          DO 10 I=1,6
          DO 10 J=1,6
     10   BME(I,J)=CF*WF(I,J)
C
          BME(1,1)=CM+BME(1,1)
          BME(2,2)=CM+BME(2,2)
          BME(3,3)=CM+BME(3,3)
          BME(4,4)=CM+BME(4,4)
          BME(5,5)=CM+BME(5,5)
          BME(6,6)=CM+BME(6,6)
C
          CALL AIINV(BME)
C
          RETURN
          END
C ======================================================================
C ======================== B E C O N  ==========================
C ======================================================================
          SUBROUTINE BECON(CM,CF)
          IMPLICIT REAL*8 (A-H,O-Z)
C
          CALL QCONCEN
          CALL WFCON
          CALL BMECON(CM,CF)
          CALL BFECON
C
          RETURN
          END
C ======================================================================
C ======================== B P C O N  ==========================
C ======================================================================
          SUBROUTINE BPCON(PSMB,CM,CF)
          IMPLICIT REAL*8 (A-H,O-Z)
          DIMENSION PSMB(6,6)
C
          CALL PQCONCEN(PSMB)
          CALL PWFCON(PSMB)
          CALL BMECON(CM,CF)
          CALL BFECON
C
          RETURN
          END
C ======================================================================
C ======================== P Q C O N C E N ==========================
C ======================================================================
          SUBROUTINE PQCONCEN(PSMB)
          IMPLICIT REAL*8 (A-H,O-Z)
          COMMON/QCON1/Q(6,6)
          COMMON/PCON1/P(6,6)
          DIMENSION PSMB(6,6)
          DIMENSION EP(6,6),Q1(6,6)
C
          CALL ATIMB(EP,PSMB,P)
          CALL ATIMB(Q1,EP,PSMB)
          CALL SUBMT(Q,PSMB,Q1)
C
          RETURN
          END
C
C ======================================================================
C ======================== P W F C O N  ==========================
C ======================================================================
          SUBROUTINE PWFCON(PSMB)
```

```fortran
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/MEDAM2/ECMB(6,6),ECFB(6,6)
      COMMON/WFCON1/WF(6,6)
      COMMON/QCON1/Q(6,6)
C     COMMON/DFEC1/DFEC(6,6)
      DIMENSION DFEC(6,6),PSMB(6,6),PCMB(6,6)
C
      DO 10 I=1,6
      DO 10 J=1,6
  10  PCMB(I,J)=PSMB(I,J)
      CALL AIINV(PCMB)
C
      DO 20 I=1,6
      DO 20 J=1,6
  20  DFEC(I,J)=ECFB(I,J)-PCMB(I,J)
C
      CALL ATIMB(WF,Q,DFEC)
C
      WF(1,1)=1.DO+WF(1,1)
      WF(2,2)=1.DO+WF(2,2)
      WF(3,3)=1.DO+WF(3,3)
      WF(4,4)=1.DO+WF(4,4)
      WF(5,5)=1.DO+WF(5,5)
      WF(6,6)=1.DO+WF(6,6)
C
      CALL AIINV(WF)
C
      RETURN
      END
C ==============================================================================
C ========================= A D C O N  =========================================
C ==============================================================================
      SUBROUTINE ADCON(DM,DMM,DMF)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/AMECON1/AME(6,6)
      COMMON/AFECON1/AFE(6,6)
      DIMENSION DM(6,6),DMM(6,6),DMF(6,6)
      DIMENSION DMI(6,6)
      DIMENSION TEMP1(6,6),TEMP2(6,6)
C
      CALL DMINV(DMI,DM)
      CALL ATIMB(TEMP1,AME,DMI)
      CALL ATIMB(TEMP2,AFE,DMI)
      CALL ATIMB(AME,DMM,TEMP1)
      CALL ATIMB(AFE,DMF,TEMP2)
C
      RETURN
      END
C
C ==============================================================================
C ========================= B D C O N  =========================================
C ==============================================================================
      SUBROUTINE BDCON(DM,DMM,DMF)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/BMECON1/BME(6,6)
      COMMON/BFECON1/BFE(6,6)
      DIMENSION DM(6,6),DMM(6,6),DMF(6,6)
      DIMENSION DMMI(6,6),DMFI(6,6)
      DIMENSION TEMP1(6,6),TEMP2(6,6)
C
      CALL DMINV(DMMI,DMM)
      CALL DMINV(DMFI,DMF)
      CALL ATIMB(TEMP1,BME,DM)
      CALL ATIMB(TEMP2,BFE,DM)
      CALL ATIMB(BME,DMMI,TEMP1)
      CALL ATIMB(BFE,DMFI,TEMP2)
C
      RETURN
      END
C
C ==============================================================================
C ========================= O V D M G ==========================================
C ==============================================================================
      SUBROUTINE OVDMG(DM,DMM,DMF,BME,BFE,CM,CF)
C
C THIS SUBPROGRAM CALCULATES THE OVERALL DAMAGE EFFECT TENSOR M(I,J)
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION DM(6,6),DMM(6,6),DMF(6,6),BME(6,6),BFE(6,6)
      DIMENSION TEMP1(6,6),TEMP2(6,6)
```

123

```fortran
      CALL ATIMB(TEMP1,DMM,BME)
      CALL ATIMB(TEMP2,DMF,BFE)
      DO 10 I=1,6
      DO 10 J=1,6
   10 DM(I,J)=CM*TEMP1(I,J)+CF*TEMP2(I,J)
      RETURN
      END
C ================================================================
C ====================== V O L F R C =============================
C ================================================================
      SUBROUTINE VOLFRC(CM,CF,PHIM,PHIF,CMO,CFO)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION PHIM(6),PHIF(6)
C
      CM1=(1.DO-PHIM(1))/((1.DO-PHIM(1))+(1.DO-PHIF(1))*CFO/CMO)
      CF1=(1.DO-PHIF(1))/((1.DO-PHIM(1))*CMO/CFO+(1.DO-PHIF(1)))
      CM2=(1.DO-PHIM(2))/((1.DO-PHIM(2))+(1.DO-PHIF(2))*CFO/CMO)
      CF2=(1.DO-PHIF(2))/((1.DO-PHIM(2))*CMO/CFO+(1.DO-PHIF(2)))
      CM3=(1.DO-PHIM(4))/((1.DO-PHIM(4))+(1.DO-PHIF(4))*CFO/CMO)
      CF3=(1.DO-PHIF(4))/((1.DO-PHIM(4))*CMO/CFO+(1.DO-PHIF(4)))
C
      CM=(CM1+CM2+CM3)/3.DO
      CF=(CF1+CF2+CF3)/3.DO
      RETURN
      END
C ================================================================
C ====================== T R N S T F =============================
C ================================================================
      SUBROUTINE TRNSTF(D,THETA)
C
C THIS SUBPROGRAM CALCULATES THE LAMINAE ELASTIC STIFFNESS
C MATRIX IN THE LOADING DIRECTION
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION D(6,6),TP(6,6)
      DIMENSION T(6,6),TT(6,6)
C
      PI=4.0DO*DATAN(1.0DO)
      RADIAN=THETA*PI/180.0DO
      C=DCOS(RADIAN)
      S=DSIN(RADIAN)
C
      T(1,1)=C*C
      T(1,2)=S*S
      T(1,4)=-2.DO*C*S
      T(2,1)=S*S
      T(2,2)=C*C
      T(2,4)=2.DO*C*S
      T(3,3)=1.DO
      T(4,1)=C*S
      T(4,2)=-C*S
      T(4,4)=C*C-S*S
      T(5,5)=C
      T(5,6)=S
      T(6,5)=-S
      T(6,6)=C
C
      TT(1,1)=C*C
      TT(1,2)=S*S
      TT(1,4)=C*S
      TT(2,1)=S*S
      TT(2,2)=C*C
      TT(2,4)=-C*S
      TT(3,3)=1.DO
      TT(4,1)=-2.DO*C*S
      TT(4,2)=2.DO*C*S
      TT(4,4)=C*C-S*S
      TT(5,5)=C
      TT(5,6)=-S
      TT(6,5)=S
      TT(6,6)=C
C
      CALL ATIMB(TP,T,D)
      CALL ATIMB(D,TP,TT)
C
      RETURN
      END
C ================================================================
C ====================== E F E C T 1 =============================
C ================================================================
```

124

```fortran
      SUBROUTINE EFECT1(SB,DM,S)
C ======================================================================
C I   COMPUTE EFFECTIVE STRESS                                        I
C I                                                                   I
C ======================================================================
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION SB(6),DM(6,6),S(6)
      CALL MULV1(SB,DM,S)
      RETURN
      END
C ======================================================================
C ======================= E F E C T 2 ==================================
C ======================================================================
      SUBROUTINE EFECT2(DEB,DM,DE)
C ======================================================================
C I   COMPUTE EFFECTIVE STRAIN                                        I
C ======================================================================
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION DEB(6),DM(6,6),DMI(6,6),DE(6)
      CALL DMINV(DMI,DM)
      CALL MULV1(DEB,DMI,DE)
      RETURN
      END
C ======================================================================
C ======================= E F E C T 3 ==================================
C ======================================================================
      SUBROUTINE EFECT3(BTDS,TDS,DMP,DM,DPHI,TSTRS)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION BTDS(6),TDS(6),DMP(6,6,6),DM(6,6),DPHI(6),TSTRS(6)
      DIMENSION XM(6,6),TEMP1(6),TEMP2(6)
      DO 10 I=1,6
      DO 10 J=1,6
      XM(I,J)=0.D0
      DO 10 K=1,6
   10 XM(I,J)=XM(I,J)+DMP(I,J,K)*DPHI(K)
      CALL MULV1(TEMP1,XM,TSTRS)
      CALL MULV1(TEMP2,DM,TDS)
      DO 20 I=1,6
   20 BTDS(I)=TEMP1(I)+TEMP2(I)
      RETURN
      END
C ======================================================================
C ======================= D E L P L D  =================================
C ======================================================================
      SUBROUTINE DELPLD(DP,DBAR,EI,DM,DMP,S)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION DP(6,6),DBAR(6,6),EI(6,6)
      DIMENSION DM(6,6),DMI(6,6),DMP(6,6,6),T(6,6),S(6)
      DIMENSION TEMP(6,6),XO(6,6)
C
      CALL XODMG(XO,DBAR,DM,DMP,T,EI,S)
      CALL AIINV(XO)
      CALL DMINV(DMI,DM)
      CALL ATIMB(TEMP,DBAR,DMI)
      CALL ATIMB(DP,XO,TEMP)
C
      RETURN
      END
C ======================================================================
C ======================= X O D M G  ===================================
C ======================================================================
      SUBROUTINE XODMG(XO,DBAR,DM,DMP,T,EI,S)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION XO(6,6),DBAR(6,6),EI(6,6),DM(6,6),DMP(6,6,6),T(6,6)
      DIMENSION S(6),XA1(6,6),XA2(6,6),TEMP(6)
      DIMENSION X(6,6,6),Z(6,6,6)
C
      CALL XZDMG(X,Z,T,DM,DMP)
C
      DO 10 I=1,6
      DO 10 J=1,6
      XO(I,J)=0.D0
      DO 10 K=1,6
   10 XO(I,J)=XO(I,J)+X(I,J,K)*S(K)
C
      CALL MULV1(TEMP,EI,S)
C
      DO 20 I=1,6
      DO 20 J=1,6
      XA1(I,J)=0.D0
```

125

```
      DO 20 K=1,6
 20   XA1(I,J)=XA1(I,J)+Z(I,J,K)*TEMP(K)
C
      CALL ATIMB(XA2,DBAR,XA1)
      CALL SUBMT(XA1,XA,XA2)
      DO 30 I=1,6
      DO 30 J=1,6
 30   XA1(I,J)=XA1(I,J)
      CALL ADMTX(XO,XA1,DM)
      CALL TRANP(XO)
C
      RETURN
      END
C =======================================================================
C ========================= X Z D M G ===============================
C =======================================================================
      SUBROUTINE XZDMG(X,Z,T,DM,DMP)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION X(6,6,6),Z(6,6,6),T(6,6),DM(6,6),DMP(6,6,6)
      DIMENSION DMIP(6,6,6)
C
      DO 10 I=1,6
      DO 10 J=1,6
      DO 10 K=1,6
      X(I,J,K)=0.DO
      DO 10 L=1,6
 10   X(I,J,K)=X(I,J,K)+DMP(I,J,L)*T(L,K)
C
      CALL DMIPHI(DMIP,DMP,DM)
C
      DO 20 I=1,6
      DO 20 J=1,6
      DO 20 K=1,6
      Z(I,J,K)=0.DO
      DO 20 L=1,6
 20   Z(I,J,K)=Z(I,J,K)+DMIP(I,J,L)*T(L,K)
      RETURN
      END
C =======================================================================
C ======================= D M I P H I ===============================
C =======================================================================
      SUBROUTINE DMIPHI(DMIP,DMP,DM)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION DMIP(6,6,6),DMP(6,6,6),DM(6,6),DMI(6,6),TEMP(6,6,6)
C
      CALL DMINV(DMI,DM)
C
      DO 10 M=1,6
      DO 10 J=1,6
      DO 10 K=1,6
      TEMP(M,J,K)=0.DO
      DO 10 L=1,6
 10   TEMP(M,J,K)=TEMP(M,J,K)+DMP(M,L,K)*DMI(L,J)
C
      DO 20 I=1,6
      DO 20 J=1,6
      DO 20 K=1,6
      DMIP(I,J,K)=0.DO
      DO 20 M=1,6
 20   DMIP(I,J,K)=DMIP(I,J,K)+DMI(M,I)*TEMP(M,J,K)
C
      DO 30 I=1,6
      DO 30 J=1,6
      DO 30 K=1,6
 30   DMIP(I,J,K)=-DMIP(I,J,K)
C
      RETURN
      END
C =======================================================================
C ======================= E L D A M G ===============================
C =======================================================================
      SUBROUTINE ELDAMG(ED,DM,E)
C
C THIS SUBPROGRAM CALCULATES THE DAMAGED ELASTIC STIFFNESS FOR
C MATRIX OR FIBERS
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION ED(6,6),DM(6,6),E(6,6)
      DIMENSION DMI(6,6),DMIT(6,6),DMIE(6,6)
      CALL DMINV(DMI,DM)
```

126

```
      CALL TRANR(DMIT,DMI)
      CALL ATIMB(DMIE,DMI,E)
      CALL ATIMB(ED,DMIE,DMIT)
      RETURN
      END
C =====================================================================
C ====================== E I D A M G ==================================
C =====================================================================
C
      SUBROUTINE EIDAMG(EID,DM,EIB)
C
C THIS SUBPROGRAM CALCULATES THE DAMAGED ELASTIC COMPLIANCE
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION EID(6,6),DM(6,6),EIB(6,6)
      DIMENSION DMT(6,6),DMTE(6,6)
C
      CALL TRANR(DMT,DM)
      CALL ATIMB(DMTE,DMT,EIB)
      CALL ATIMB(EID,DMTE,DM)
C
      RETURN
      END
C =====================================================================
C ======================= D M G C R T =================================
C =====================================================================
      SUBROUTINE DMGCRT(G,Y,HI)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION Y(6),HI(6)
      CALL SCLVV(G,HI,Y)
      G=G*G
      RETURN
      END
C =====================================================================
C ========================= G D E R ===================================
C =====================================================================
      SUBROUTINE GDER(GY,GP,YP,PHI,Y,HI,A1,A2,A3,B1,B2,B3)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION GY(6),GP(6),YP(6,6),PHI(6),Y(6),HI(6)
      DIMENSION GH(6),HD(6,6),GP1(6),GP2(6)
      DIMENSION XD(6,6),H1D(6),H2D(6),H3D(6),H4D(6),H5D(6),H6D(6)
      DIMENSION X(6),DDT(6),HH(6,6)
C
      CALL SCLVV(A,HI,Y)
      DO 10 I=1,6
      GY(I)=2.D0*A*HI(I)
  10  GH(I)=2.D0*A*Y(I)
C
      H1=(A1*A1*PHI(1)+B1)
      H2=(A2*A2*PHI(2)+B2)
      H3=(A3*A3*PHI(3)+B3)
      H4=(A1*A2*PHI(4))
      H5=(A2*A3*PHI(5))
      H6=(A1*A3*PHI(6))
C
      H1D(1)=A1*A1+B1
      H2D(2)=A2*A2-B2
      H3D(3)=A3*A3-B3
      H4D(4)=A1*A2+((B1*B2)**0.5D0)
      H5D(5)=A2*A3+((B2*B3)**0.5D0)
      H6D(6)=A1*A3+((B1*B3)**0.5D0)
C
      DT=H1*H2*H3+2.D0*H4*H5*H6-H1*H5*H5-H2*H6*H6-H3*H4*H4
      DDT(1)=H1D(1)*H2*H3+H1*H2D(1)*H3+H1*H2*H3D(1)+
     $2.D0*(H4D(1)*H5*H6)-(H1D(1)*H5*H5+2.D0*H1*H5*H5D(1))-
     $(H2D(1)*H6*H6+2.D0*H2*H6*H6D(1))-
     $(H3D(1)*H4*H4+2.D0*H3*H4*H4D(1))
      DDT(2)=H1D(2)*H2*H3+H1*H2D(2)*H3+H1*H2*H3D(2)+
     $2.D0*(H4D(2)*H5*H6)-(H1D(2)*H5*H5+2.D0*H1*H5*H5D(2))-
     $(H2D(2)*H6*H6+2.D0*H2*H6*H6D(2))-
     $(H3D(2)*H4*H4+2.D0*H3*H4*H4D(2))
      DDT(3)=H1D(3)*H2*H3+H1*H2D(3)*H3+H1*H2*H3D(3)+
     $2.D0*(H4D(3)*H5*H6)-(H1D(3)*H5*H5+2.D0*H1*H5*H5D(3))-
     $(H2D(3)*H6*H6+2.D0*H2*H6*H6D(3))-
     $(H3D(3)*H4*H4+2.D0*H3*H4*H4D(3))
      DDT(4)=H1D(4)*H2*H3+H1*H2D(4)*H3+H1*H2*H3D(4)+
     $2.D0*(H4D(4)*H5*H6)-(H1D(4)*H5*H5+2.D0*H1*H5*H5D(4))-
     $(H2D(4)*H6*H6+2.D0*H2*H6*H6D(4))-
     $(H3D(4)*H4*H4+2.D0*H3*H4*H4D(4))
      DDT(5)=H1D(5)*H2*H3+H1*H2D(5)*H3+H1*H2*H3D(5)+
```

```fortran
     $2.D0*(H4D(5)*H5*H6)-(H1D(5)*H5*H5+2.D0*H1*H5*H5D(5))-
     $(H2D(5)*H6*H6+2.D0*H2*H6*H6D(5))-
     $(H3D(5)*H4*H4+2.D0*H3*H4*H4D(5))
      DDT(6)=H1D(6)*H2*H3+H1*H2D(6)*H3+H1*H2*H3D(6)+
     $2.D0*(H4D(6)*H5*H6)-(H1D(6)*H5*H5+2.D0*H1*H5*H5D(6))-
     $(H2D(6)*H6*H6+2.D0*H2*H6*H6D(6))-
     $(H3D(6)*H4*H4+2.D0*H3*H4*H4D(6))
C
      XD(1,1)=H2D(1)*H3+H2*H3D(1)-2.D0*H5*H5D(1)
      XD(1,2)=H2D(2)*H3+H2*H3D(2)-2.D0*H5*H5D(2)
      XD(1,3)=H2D(3)*H3+H2*H3D(3)-2.D0*H5*H5D(3)
      XD(1,4)=H2D(4)*H3+H2*H3D(4)-2.D0*H5*H5D(4)
      XD(1,5)=H2D(5)*H3+H2*H3D(5)-2.D0*H5*H5D(5)
      XD(1,6)=H2D(6)*H3+H2*H3D(6)-2.D0*H5*H5D(6)
      XD(2,1)=H1D(1)*H3+H1*H3D(1)-2.D0*H6*H6D(1)
      XD(2,2)=H1D(2)*H3+H1*H3D(2)-2.D0*H6*H6D(2)
      XD(2,3)=H1D(3)*H3+H1*H3D(3)-2.D0*H6*H6D(3)
      XD(2,4)=H1D(4)*H3+H1*H3D(4)-2.D0*H6*H6D(4)
      XD(2,5)=H1D(5)*H3+H1*H3D(5)-2.D0*H6*H6D(5)
      XD(2,6)=H1D(6)*H3+H1*H3D(6)-2.D0*H6*H6D(6)
      XD(3,1)=H1D(1)*H2+H1*H2D(1)-2.D0*H4*H6D(1)
      XD(3,2)=H1D(2)*H2+H1*H2D(2)-2.D0*H4*H5D(2)
      XD(3,3)=H1D(3)*H2+H1*H2D(3)-2.D0*H4*H5D(3)
      XD(3,4)=H1D(4)*H2+H1*H2D(4)-2.D0*H4*H5D(4)
      XD(3,5)=H1D(5)*H2+H1*H2D(5)-2.D0*H4*H5D(5)
      XD(3,6)=H1D(6)*H2+H1*H2D(6)-2.D0*H4*H5D(6)
      XD(4,1)=H5D(1)*H6+H5*H6D(1)-H3D(1)*H4-H3*H4D(1)
      XD(4,2)=H5D(2)*H6+H5*H6D(2)-H3D(2)*H4-H3*H4D(2)
      XD(4,3)=H5D(3)*H6+H5*H6D(3)-H3D(3)*H4-H3*H4D(3)
      XD(4,4)=H5D(4)*H6+H5*H6D(4)-H3D(4)*H4-H3*H4D(4)
      XD(4,5)=H5D(5)*H6+H5*H6D(5)-H3D(5)*H4-H3*H4D(5)
      XD(4,6)=H5D(6)*H6+H5*H6D(6)-H3D(6)*H4-H3*H4D(6)
      XD(5,1)=H4D(1)*H6+H4*H6D(1)-H1D(1)*H5-H1*H5D(1)
      XD(5,2)=H4D(2)*H6+H4*H6D(2)-H1D(2)*H5-H1*H5D(2)
      XD(5,3)=H4D(3)*H6+H4*H6D(3)-H1D(3)*H5-H1*H5D(3)
      XD(5,4)=H4D(4)*H6+H4*H6D(4)-H1D(4)*H5-H1*H5D(4)
      XD(5,5)=H4D(5)*H6+H4*H6D(5)-H1D(5)*H5-H1*H5D(5)
      XD(5,6)=H4D(6)*H6+H4*H6D(6)-H1D(6)*H5-H1*H5D(6)
      XD(6,1)=H4D(1)*H5+H4*H5D(1)-H2D(1)*H6-H2*H6D(1)
      XD(6,2)=H4D(2)*H5+H4*H5D(2)-H2D(2)*H6-H2*H6D(2)
      XD(6,3)=H4D(3)*H5+H4*H5D(3)-H2D(3)*H6-H2*H6D(3)
      XD(6,4)=H4D(4)*H5+H4*H5D(4)-H2D(4)*H6-H2*H6D(4)
      XD(6,5)=H4D(5)*H5+H4*H5D(5)-H2D(5)*H6-H2*H6D(5)
      XD(6,6)=H4D(6)*H5+H4*H5D(6)-H2D(6)*H6-H2*H6D(6)
C
      X(1)=H2*H3-H5*H5
      X(2)=H1*H3-H6*H6
      X(3)=H1*H2-H4*H4
      X(4)=H5*H6-H3*H4
      X(5)=H4*H6-H1*H5
      X(6)=H4*H5-H2*H6
C
      DO 20 I=1,6
      DO 20 J=1,6
   20 HD(I,J)=(XD(I,J)*DT-X(I)*DDT(J))/(DT*DT)
C
      CALL MULV1(GP1,HD,GH)
      CALL MULV1(GP2,YP,GY)
      DO 30 I=1,6
   30 GP(I)=(GP1(I)+GP2(I))
C
      RETURN
      END
C ======================================================================
C ====================== Y D E R =======================================
C ======================================================================
      SUBROUTINE YDER(YSIG,YPHI,PHI,DM,DMP,S,EI)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION YSIG(6,6),YPHI(6,6),PHI(6),DM(6,6),DMP(6,6,6)
      DIMENSION S(6),EI(6,6),DMT(6,6)
      DIMENSION YS1(6),YS2(6),YS3(6),YS4(6)
      DIMENSION X(6,6),XT(6,6),TEMP(6)
      DIMENSION DMP2(6,6,6,6),X1(6,6),YP1(6,6),YP2(6,6)
      DIMENSION T1(6,6),TT1(6,6),T2(6,6),TT2(6,6)
C
      CALL ATIMB(T1,EI,DM)
      CALL TRANR(TT1,T1)
      CALL TRANR(DMT,DM)
      CALL ATIMB(T2,DMT,EI)
      CALL TRANR(TT2,T2)
```

128

```fortran
C
      DO 10 IJ=1,6
C
      DO 15 I=1,6
      DO 15 J=1,6
  15  X(I,J)=DMP(I,J,IJ)
      CALL TRANR(XT,X)
C
      CALL MULV1(TEMP,T1,S)
      CALL MULV1(YS1,XT,TEMP)
C
      CALL MULV1(TEMP,X,S)
      CALL MULV1(YS2,TT1,TEMP)
C
      CALL MULV1(TEMP,X,S)
      CALL MULV1(YS3,T2,TEMP)
C
      CALL MULV1(TEMP,TT2,S)
      CALL MULV1(YS4,XT,TEMP)
C
      DO 20 K=1,6
  20  YSIG(K,IJ)=0.5DO*(YS1(K)+YS2(K)+YS3(K)+YS4(K))
C
  10  CONTINUE
C
      CALL MDPH2(DMP2,PHI)
C
      DO 40 I=1,6
      DO 50 J=1,6
C
      DO 60 K=1,6
      DO 60 L=1,6
  60  X(K,L)=DMP2(K,L,I,J)
C
      CALL MULV1(YS1,X,S)
      CALL MULV1(YS2,YS1,EI)
      CALL MULV1(YS1,DM,S)
      CALL SCLVV(A1,YS2,YS1)
C
      CALL MULV1(YS1,DM,S)
      CALL MULV1(YS2,YS1,EI)
      CALL MULV1(YS3,X,S)
      CALL SCLVV(B1,YS2,YS3)
C
      YP1(I,J)=0.5DO*(A1+B1)
  50  CONTINUE
  40  CONTINUE
C
      DO 70 I=1,6
C
      DO 75 K1=1,6
      DO 75 L1=1,6
  75  X(K1,L1)=DMP(K1,L1,I)
C
      DO 80 J=1,6
C
      DO 85 K2=1,6
      DO 85 L2=1,6
  85  X1(K2,L2)=DMP(K2,L2,J)
C
      CALL MULV1(YS1,X,S)
      CALL MULV1(YS2,X1,S)
      CALL MULV1(YS3,EI,YS2)
      CALL SCLVV(A2,YS1,YS3)
C
      CALL MULV1(YS1,X1,S)
      CALL MULV1(YS2,YS1,EI)
      CALL MULV1(YS3,X,S)
      CALL SCLVV(B2,YS2,YS3)
C
      YP2(I,J)=0.5DO*(A2+B2)
  80  CONTINUE
  70  CONTINUE
C
      CALL ADMTX(YPHI,YP1,YP2)
C
      RETURN
      END
C =================================================================
C ======================= D M G H A R ============================
```

```
C =========================================================================
      SUBROUTINE DMGHAR(HI,A1,A2,A3,B1,B2,B3,PHI,HK,R1,R2,R3,Q1,Q2,Q3
     $,U1,U2,U3,V1,V2,V3)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION HI(6),PHI(6)
C
      A1=DSQRT(U1*Q1*((HK/U1)**R1))
      A2=DSQRT(U2*Q2*((HK/U2)**R2))
      A3=DSQRT(U3*Q3*((HK/U3)**R3))
C
      B1=U1*V1*V1
      B2=U2*V2*V2
      B3=U3*V3*V3
C
      H1=(A1*A1*PHI(1)+B1)
      H2=(A2*A2*PHI(2)+B2)
      H3=(A3*A3*PHI(3)+B3)
      H4=(A1*A2*PHI(4))
      H5=(A2*A3*PHI(5))
      H6=(A1*A3*PHI(6))
C
      DT=H1*H2*H3+2.DO*H4*H5*H6-H1*H5*H5-H2*H6*H6-H3*H4*H4
      HI(1)=(H2*H3-H5*H5)/DT
      HI(2)=(H1*H3-H6*H6)/DT
      HI(3)=(H1*H2-H4*H4)/DT
      HI(4)=(H5*H6-H3*H4)/DT
      HI(5)=(H4*H6-H1*H5)/DT
      HI(6)=(H4*H5-H2*H6)/DT
      RETURN
      END
C =========================================================================
C ======================= D M G F R C =====================================
C =========================================================================
      SUBROUTINE DMGFRC(Y,DM,DMP,S,EI)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION Y(6),DM(6,6),DMP(6,6,6),S(6),EI(6,6)
      DIMENSION X(6,6),TEMP1(6),TEMP2(6)
      DO 10 K=1,6
      DO 20 I=1,6
      DO 20 J=1,6
   20 X(I,J)=DMP(I,J,K)
      CALL MULV1(TEMP1,DM,S)
      CALL MULV1(TEMP2,EI,TEMP1)
      CALL MULV1(TEMP1,X,S)
      CALL SCLVV(A,TEMP1,TEMP2)
      CALL MULV1(TEMP1,X,S)
      CALL MULV1(TEMP2,EI,TEMP1)
      CALL MULV1(TEMP1,DM,S)
      CALL SCLVV(B,TEMP1,TEMP2)
      Y(K)=0.5DO*(A+B)
   10 CONTINUE
      RETURN
      END
C =========================================================================
C ======================= D M G H K =====================================
C =========================================================================
      SUBROUTINE DMGHK(HK,Y,DPHI)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION Y(6),DPHI(6)
      CALL SCLVV(DHK,Y,DPHI)
      HK=HK+DHK
      RETURN
      END
C =========================================================================
C ================     C A L D M G     ===========================
C =========================================================================
      SUBROUTINE CALDMG(PHI,DM,XMP,S,DS,EI,HK,Y,HI,A1,A2,A3
     $,B1,B2,B3,CFAIL)
C
C THIS SUBROUTINE CALCULATES THE DAMAGE VARIABLES
C
      IMPLICIT REAL*8 (A-H,O-Z)
      CHARACTER*1 CFAIL
      DIMENSION PHI(6),D(6,6),DM(6,6),XMP(6,6,6)
      DIMENSION S(6),DS(6),EI(6,6),Y(6),HI(6)
      DIMENSION GY(6),GP(6)
      DIMENSION A(6,6),B(6,6),C(6,6),AC(6,6),TEMP(6,6)
C
      CALL YDER(B,C,PHI,DM,XMP,S,EI)
      CALL GDER(GY,GP,C,PHI,Y,HI,A1,A2,A3,B1,B2,B3)
```

130

```
          CALL SCLVV(GPY,GP,GY)
          DO 20 I=1,6
          DO 20 J=1,6
   20     A(I,J)=GY(I)*GY(J)/GPY
          CALL ATIMB(AC,A,C)
          DO 30 I=1,6
          DO 30 J=1,6
          IF (I.EQ.J) THEN
            D(I,J)=1.D0-AC(I,J)
          ELSE
            D(I,J)=-AC(I,J)
          END IF
   30     CONTINUE
          CALL AIINV(D)
          CALL ATIMB(TEMP,D,A)
          CALL ATIMB(D,TEMP,B)
          CALL TRANP(D)
          CALL MULV1(DPHI,D,DS)
          DO 40 I=1,6
   40     PHI(I)=PHI(I)+DPHI(I)
C
          CALL DMGHK(HK,Y,DPHI)
          CALL RUPTR(PHI,CFAIL)
          RETURN
          END
C =====================================================================
C ======================= R U P T R ==================================
C =====================================================================
C
          SUBROUTINE RUPTR(PHI,IRUP)
C
          IMPLICIT REAL*8 (A-H,O-Z)
          CHARACTER*1 IRUP
          DIMENSION PHI(6)
C
          PHIMAX1=0.5D0*(PHI(1)+PHI(2))
          PHIMAX2=PHI(1)-PHI(2)
          PHIMAX3=PHI(4)*PHI(4)
          PHIMAX=PHIMAX1+(PHIMAX2*PHIMAX2*0.25D0+PHIMAX3)**0.5D0
C
          IF(PHIMAX.GT.0.99) THEN
            PHI(1)=0.9
            PHI(2)=0.9
            PHI(3)=0.9
            PHI(4)=0.9
            PHI(5)=0.9
            PHI(6)=0.9
          END IF
C
          RETURN
          END
C =====================================================================
C ======================= T R A N R ==================================
C =====================================================================
          SUBROUTINE TRANR(B,A)
C
C THIS SUBPROGRAM CALCULATES THE TRANSPOSE OF A MATRIX "A"
C STORING IT IN THE ARRAY "B"
C
          IMPLICIT REAL*8 (A-H,O-Z)
          DIMENSION A(6,6),B(6,6)
C
          DO 10 I=1,6
          DO 10 J=1,6
   10     B(J,I)=A(I,J)
C
          RETURN
          END
C =====================================================================
C ======================= T R A N P ==================================
C =====================================================================
          SUBROUTINE TRANP(A)
C
C THIS SUBPROGRAM CALCULATES THE TRANSPOSE OF A SQUARE MATRIX
C "A" STORING IT IN ITSELF
C
          IMPLICIT REAL*8 (A-H,O-Z)
          DIMENSION A(6,6)
          N=6
C
```

131

```
      N1=N-1
      DO 10 I=1,N1
      I1=I+1
      DO 10 J=I1,N
      S=A(I,J)
      A(I,J)=A(J,I)
   10 A(J,I)=S
C
      RETURN
      END
C
C =====================================================================
C ========================= A D M T X ===============================
C =====================================================================
      SUBROUTINE ADMTX(C,A,B)
C
C THIS SUBPROGRAM CALCULATES THE MATRIX OPERATION
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION A(6,6),B(6,6),C(6,6)
      DO 10 I=1,6
      DO 10 J=1,6
   10 C(I,J)=A(I,J)+B(I,J)
      CONTINUE
      RETURN
      END
C =====================================================================
C ========================= S U B M T ===============================
C =====================================================================
      SUBROUTINE SUBMT(C,A,B)
C
C THIS SUBPROGRAM CALCULATES THE MATRIX OPERATION
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION A(6,6),B(6,6),C(6,6)
      DO 10 I=1,6
      DO 10 J=1,6
      C(I,J)=A(J,I)-B(I,J)
   10 CONTINUE
      RETURN
      END
C =====================================================================
C ========================= A T I M B ===============================
C =====================================================================
      SUBROUTINE ATIMB(C,A,B)
C
C THIS SUBPROGRAM CALCULATES THE MATRIX OPERATION C = A * B
C N : ACTUAL NUMBER OF ROWS OF A AND C
C M : ACTUAL NUMBER OF COLUMNS OF A AND ROWS OF B
C L : ACTUAL NUMBER OF COLUMNS OF B AND C
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION A(6,6),B(6,6),C(6,6)
C
      DO 10 I=1,6
      DO 10 J=1,6
      C(I,J)=0.D0
      DO 10 K=1,6
   10 C(I,J)=C(I,J)+A(I,K)*B(K,J)
C
      RETURN
      END
C =====================================================================
C ========================= A T I M B 3 =============================
C =====================================================================
      SUBROUTINE ATIMB3(C,A,B)
C
C THIS SUBPROGRAM CALCULATES THE MATRIX OPERATION C = A * B
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION A(3,3),B(3,3),C(3,3)
C
      DO 10 I=1,3
      DO 10 J=1,3
      C(I,J)=0.D0
      DO 10 K=1,3
   10 C(I,J)=C(I,J)+A(I,K)*B(K,J)
C
      RETURN
      END
```

132

```
C ============================================================================
C ====================== M U L V 1 ==========================================
C ============================================================================
      SUBROUTINE MULV1(C,A,B)
C
C THIS SUBPROGRAM CALCULATES THE MATRIX OPERATION
C C =  A * B , WHERE B IS A VECTOR
C N : ACTUAL NUMBER OF ROWS A
C M : ACTUAL NUMBER OF COLUMNS OF A AND OF ROWS OF B
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION A(6,6),B(6),C(6)
      DO 10 I=1,6
      C(I)=0.DO
      DO 10 J=1,6
  10  C(I)=C(I)+A(I,J)*B(J)
      RETURN
      END
C ============================================================================
C ====================== M U L V 2 ==========================================
C ============================================================================
      SUBROUTINE MULV2(C,A,B)
C
C THIS SUBPROGRAM CALCULATES THE MATRIX OPERATION
C C =  A * B , WHERE A IS A VECTOR
C N : ACTUAL NUMBER OF COLUMNS OF A
C M : ACTUAL NUMBER OF COLUMNS OF B AND OF ROWS OF B
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION A(6),B(6,6),C(6)
      DO 10 I=1,6
      C(I)=0.DO
      DO 10 J=1,6
  10  C(I)=C(I)+A(J)*B(J,I)
      RETURN
      END
C ============================================================================
C ====================== M U L V V ==========================================
C ============================================================================
      SUBROUTINE MULVV(C,A,B)
C
C THIS SUBPROGRAM CALCULATES THE MATRIX OPERATION
C C =  A * B , WHERE A AND B IS A VECTOR
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION A(6),B(6),C(6,6)
      DO 10 I=1,6
      DO 10 J=1,6
  10  C(I,J)=A(I)*B(J)
      RETURN
      END
C ============================================================================
C ====================== S C L V V ==========================================
C ============================================================================
      SUBROUTINE SCLVV(C,A,B)
C
C THIS SUBPROGRAM CALCULATES THE MATRIX OPERATION
C C =  A * B , WHERE A AND B IS A VECTOR, AND C IS SCALAR
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION A(6),B(6)
      C=0.DO
      DO 10 I=1,6
      IF (I.GE.4) THEN
        C=C+2.DO*A(I)*B(I)
      ELSE
        C=C+A(I)*B(I)
      END IF
  10  CONTINUE
C
      RETURN
      END
C ============================================================================
C ====================== A I I N V ==========================================
C ============================================================================
      SUBROUTINE AIINV(A)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION A(6,6)
C
      N=6
```

133

```
C
      DO 26 I=1,N
      CON=A(I,I)
      DO 2 J=1,N
    2 A(I,J)=A(I,J)/CON
      A(I,I)=1.D0/CON
      DO 26 J=1,N
      IF(J-I) 4,26,4
    4 DO 36 K=1,N
      IF(K-I) 5,36,5
    5 A(J,K)=A(J,K)-A(I,K)*A(J,I)
   36 CONTINUE
      A(J,I)=-A(I,I)*A(J,I)
   26 CONTINUE
C
      RETURN
      END
C ==========================================================================
C ====================== A I I N V 3 ==================================
C ==========================================================================
      SUBROUTINE AIINV3(A)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION A(3,3)
C
      N=3
C
      DO 10 I=1,N
      CON=A(I,I)
      DO 15 J=1,N
   15 A(I,J)=A(I,J)/CON
      A(I,I)=1.D0/CON
      DO 10 J=1,N
      IF(J-I) 5,10,5
    5 DO 20 K=1,N
      IF(K-I) 6,20,6
    6 A(J,K)=A(J,K)-A(I,K)*A(J,I)
   20 CONTINUE
      A(J,I)=-A(I,I)*A(J,I)
   10 CONTINUE
C
      RETURN
      END
C ==========================================================================
C ========================= C O N V E R ==============================
C ==========================================================================
      SUBROUTINE CONVER(D2,D3)
C --------------------------------------------------------------------------
C I        THIS PROGRAM TRANSFORMS THE FOURTH ORDER STIFFNESS          I
C I        TENSOR TO A SECOND ORDER MATRIX FOR PLANE STRESS            I
C --------------------------------------------------------------------------
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION D2(6,6),D3(6,6)
C
      D2(1,1)=D3(1,1)-D3(1,3)*D3(3,1)/D3(3,3)
      D2(1,2)=D3(1,2)-D3(1,3)*D3(3,2)/D3(3,3)
      D2(1,3)=D3(1,4)-D3(1,3)*D3(3,4)/D3(3,3)
      D2(2,1)=D3(2,1)-D3(2,3)*D3(3,1)/D3(3,3)
      D2(2,2)=D3(2,2)-D3(2,3)*D3(3,2)/D3(3,3)
      D2(2,3)=D3(2,4)-D3(2,3)*D3(3,4)/D3(3,3)
      D2(3,1)=D3(4,1)-D3(4,3)*D3(3,1)/D3(3,3)
      D2(3,2)=D3(4,2)-D3(4,3)*D3(3,2)/D3(3,3)
      D2(3,3)=D3(4,4)-D3(4,3)*D3(3,4)/D3(3,3)
      D2(1,2)=(D2(1,2)+D2(2,1))*0.5D0
      D2(2,1)=D2(1,2)
      D2(1,3)=(D2(1,3)+D2(3,1))*0.5D0
      D2(3,1)=D2(1,3)
      D2(2,3)=(D2(2,3)+D2(3,2))*0.5D0
      D2(3,2)=D2(2,3)
      RETURN
      END
C ==========================================================================
C ====================== D M A G E 1 =================================
C ==========================================================================
      SUBROUTINE DMAGE1(DM,PHI)
C
C THIS SUBPROGRAM CALCULATES THE DAMAGE EFFECT TENSOR M(I,J) AND
C THE DERIVATIVE DAMAGE TENSOR WITH RESPECT TO DAMAGE VARIABLES
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION DM(6,6),PHI(6),F(6,6)
```

```
C
      P1=1.D0-PHI(1)
      P2=1.D0-PHI(2)
      P3=1.D0-PHI(3)
      P4=PHI(5)*PHI(5)*P1
      P5=PHI(6)*PHI(6)*P2
      P6=PHI(4)*PHI(4)*P3
      P7=2.D0*PHI(4)*PHI(5)*PHI(6)
      G=2.D0*(P1*P2*P3-P4-P5-P6-P7)
C
      F(1,1)=2.D0*(P2*P3-PHI(5)*PHI(5))
      F(1,4)=2.D0*(PHI(6)*PHI(5)+PHI(4)*P3)
      F(1,6)=2.D0*(PHI(4)*PHI(5)+PHI(6)*P2)
C
      F(2,2)=2.D0*(P1*P3-PHI(6)*PHI(6))
      F(2,4)=F(1,4)
      F(2,5)=2.D0*(PHI(4)*PHI(6)+PHI(5)*P1)
C
      F(3,3)=2.D0*(P1*P2-PHI(4)*PHI(4))
      F(3,5)=F(2,5)
      F(3,6)=F(1,6)
C
      F(4,1)=0.5D0*F(1,4)
      F(4,2)=F(4,1)
      F(4,4)=P2*P3+P1*P3-PHI(5)*PHI(5)-PHI(6)*PHI(6)
      F(4,5)=0.5D0*F(1,6)
      F(4,6)=0.5D0*F(2,5)
C
      F(5,2)=0.5D0*F(2,5)
      F(5,3)=F(5,2)
      F(5,4)=0.5D0*F(1,6)
      F(5,5)=P1*P3+P1*P2-PHI(6)*PHI(6)-PHI(4)*PHI(4)
      F(5,6)=0.5D0*F(1,4)
C
      F(6,1)=F(5,4)
      F(6,3)=F(6,1)
      F(6,4)=F(5,2)
      F(6,5)=F(5,6)
      F(6,6)=P2*P3+P1*P2-PHI(5)*PHI(5)-PHI(4)*PHI(4)
C
      DO 10 I=1,6
      DO 10 J=1,6
  10  DM(I,J)=F(I,J)/G
C
      RETURN
      END
C =============================================================================
C ======================= D M A G E 2 =========================================
C =============================================================================
      SUBROUTINE DMAGE2(DM,XMPHI,PHI)
C
C THIS SUBPROGRAM CALCULATES THE DAMAGE EFFECT TENSOR M(I,J) AND
C THE DERIVATIVE DAMAGE TENSOR WITH RESPECT TO DAMAGE VARIABLES
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION DM(6,6),XMPHI(6,6,6),PHI(6),F(6,6),F1(6,6,6)
      DIMENSION G1(6)
C
      P1=1.D0-PHI(1)
      P2=1.D0-PHI(2)
      P3=1.D0-PHI(3)
      P4=PHI(5)*PHI(5)*P1
      P5=PHI(6)*PHI(6)*P2
      P6=PHI(4)*PHI(4)*P3
      P7=2.D0*PHI(4)*PHI(5)*PHI(6)
      G=2.D0*(P1*P2*P3-P4-P5-P6-P7)
C
      G1(1)=-2.D0*(P2*P3-PHI(5)*PHI(5))
      G1(2)=-2.D0*(P1*P3-PHI(6)*PHI(6))
      G1(3)=-2.D0*(P1*P2-PHI(4)*PHI(4))
      G1(4)=-4.D0*(PHI(4)*P3+PHI(5)*PHI(6))
      G1(5)=-4.D0*(PHI(5)*P1+PHI(4)*PHI(6))
      G1(6)=-4.D0*(PHI(6)*P2+PHI(4)*PHI(5))
C
      F(1,1)=2.D0*(P2*P3-PHI(5)*PHI(5))
      F(1,4)=2.D0*(PHI(6)*PHI(5)+PHI(4)*P3)
      F(1,6)=2.D0*(PHI(4)*PHI(5)+PHI(6)*P2)
C
      F(2,2)=2.D0*(P1*P3-PHI(6)*PHI(6))
      F(2,4)=F(1,4)
```

135

```
      F(2,5)=2.DO*(PHI(4)*PHI(6)+PHI(5)*P1)
C
      F(3,3)=2.DO*(P1*P2-PHI(4)*PHI(4))
      F(3,5)=F(2,5)
      F(3,6)=F(1,6)
C
      F(4,1)=0.5DO*F(1,4)
      F(4,2)=F(4,1)
      F(4,4)=P2*P3+P1*P3-PHI(5)*PHI(5)-PHI(6)*PHI(6)
      F(4,5)=0.5DO*F(1,6)
      F(4,6)=0.5DO*F(2,5)
C
      F(5,2)=0.5DO*F(2,5)
      F(5,3)=F(5,2)
      F(5,4)=0.5DO*F(1,6)
      F(5,5)=P1*P3+P1*P2-PHI(6)*PHI(6)-PHI(4)*PHI(4)
      F(5,6)=0.5DO*F(1,4)
C
      F(6,1)=F(5,4)
      F(6,3)=F(6,1)
      F(6,4)=F(5,2)
      F(6,5)=F(5,6)
      F(6,6)=P2*P3+P1*P2-PHI(5)*PHI(5)-PHI(4)*PHI(4)
C
      DO 10 I=1,6
      DO 10 J=1,6
  10  DM(I,J)=F(I,J)/G
C
C
      F1(1,1,2)=2.DO*(-P3)
      F1(1,1,3)=2.DO*(-P2)
      F1(1,1,5)=-4.DO*PHI(5)
C
      F1(1,4,3)=2.DO*(-PHI(4))
      F1(1,4,4)=2.DO*P3
      F1(1,4,5)=2.DO*PHI(6)
      F1(1,4,6)=2.DO*PHI(5)
C
      F1(1,6,2)=2.DO*(-PHI(6))
      F1(1,6,4)=2.DO*PHI(5)
      F1(1,6,5)=2.DO*PHI(4)
      F1(1,6,6)=2.DO*P2
C
C
      F1(2,2,1)=2.DO*(-P3)
      F1(2,2,3)=2.DO*(-P1)
      F1(2,2,6)=-4.DO*PHI(6)
C
      F1(2,4,3)=F1(1,4,3)
      F1(2,4,4)=F1(1,4,4)
      F1(2,4,5)=F1(1,4,5)
      F1(2,4,6)=F1(1,4,6)
C
      F1(2,5,1)=2.DO*(-PHI(5))
      F1(2,5,4)=2.DO*PHI(6)
      F1(2,5,5)=2.DO*P1
      F1(2,5,6)=2.DO*PHI(4)
C
C
      F1(3,3,1)=2.DO*(-P2)
      F1(3,3,2)=2.DO*(-P1)
      F1(3,3,4)=-4.DO*PHI(4)
C
      F1(3,5,1)=F1(2,5,1)
      F1(3,5,4)=F1(2,5,4)
      F1(3,5,5)=F1(2,5,5)
      F1(3,5,6)=F1(2,5,6)
C
      F1(3,6,2)=F1(1,6,2)
      F1(3,6,4)=F1(1,6,4)
      F1(3,6,5)=F1(1,6,5)
      F1(3,6,6)=F1(1,6,6)
C
C
      F1(4,1,3)=0.5DO*F1(1,4,3)
      F1(4,1,4)=0.5DO*F1(1,4,4)
      F1(4,1,5)=0.5DO*F1(1,4,5)
      F1(4,1,6)=0.5DO*F1(1,4,6)
C
      F1(4,2,3)=F1(4,1,3)
```

```
      F1(4,2,4)=F1(4,1,4)
      F1(4,2,5)=F1(4,1,5)
      F1(4,2,6)=F1(4,1,6)
C
      F1(4,4,1)=-P3
      F1(4,4,2)=-P3
      F1(4,4,3)=-P2-P1
      F1(4,4,5)=-2.D0*PHI(5)
      F1(4,4,6)=-2.D0*PHI(6)
C
      F1(4,5,2)=0.5D0*F1(1,6,2)
      F1(4,5,4)=0.5D0*F1(1,6,4)
      F1(4,5,5)=0.5D0*F1(1,6,5)
      F1(4,5,6)=0.5D0*F1(1,6,6)
C
      F1(4,6,1)=0.5D0*F1(2,5,1)
      F1(4,6,4)=0.5D0*F1(2,5,4)
      F1(4,6,5)=0.5D0*F1(2,5,5)
      F1(4,6,6)=0.5D0*F1(2,5,6)
C
C
      F1(5,2,1)=0.5D0*F1(2,5,1)
      F1(5,2,4)=0.5D0*F1(2,5,4)
      F1(5,2,5)=0.5D0*F1(2,5,5)
      F1(5,2,6)=0.5D0*F1(2,5,6)
C
      F1(5,3,1)=F1(5,2,1)
      F1(5,3,4)=F1(5,2,4)
      F1(5,3,5)=F1(5,2,5)
      F1(5,3,6)=F1(5,2,6)
C
      F1(5,4,2)=0.5D0*F1(1,6,2)
      F1(5,4,4)=0.5D0*F1(1,6,4)
      F1(5,4,5)=0.5D0*F1(1,6,5)
      F1(5,4,6)=0.5D0*F1(1,6,6)
C
      F1(5,5,1)=-P3-P2
      F1(5,5,2)=-P1
      F1(5,5,3)=-P1
      F1(5,5,4)=-2.D0*PHI(4)
      F1(5,5,6)=-2.D0*PHI(6)
C
      F1(5,6,3)=0.5D0*F1(1,4,3)
      F1(5,6,4)=0.5D0*F1(1,4,4)
      F1(5,6,5)=0.5D0*F1(1,4,5)
      F1(5,6,6)=0.5D0*F1(1,4,6)
C
C
      F1(6,1,2)=F1(5,4,2)
      F1(6,1,4)=F1(5,4,4)
      F1(6,1,5)=F1(5,4,5)
      F1(6,1,6)=F1(5,4,6)
C
      F1(6,3,2)=F1(6,1,2)
      F1(6,3,4)=F1(6,1,4)
      F1(6,3,5)=F1(6,1,5)
      F1(6,3,6)=F1(6,1,6)
C
      F1(6,4,1)=F1(5,2,1)
      F1(6,4,4)=F1(5,2,4)
      F1(6,4,5)=F1(5,2,5)
      F1(6,4,6)=F1(5,2,6)
C
      F1(6,5,3)=F1(5,6,3)
      F1(6,5,4)=F1(5,6,4)
      F1(6,5,5)=F1(5,6,5)
      F1(6,5,6)=F1(5,6,6)
C
      F1(6,6,1)=-P2
      F1(6,6,2)=-P3-P1
      F1(6,6,3)=-P2
      F1(6,6,4)=-2.D0*PHI(4)
      F1(6,6,5)=-2.D0*PHI(5)
C
      DO 20 I=1,6
      DO 20 J=1,6
      DO 20 K=1,6
  20  XMPHI(I,J,K)=(F1(I,J,K)*G-F(I,J)*G1(K))/(G*G)
C
      RETURN
```

```
      END
C =====================================================================
C ====================== D M I N V =====================================
C =====================================================================
      SUBROUTINE DMINV(DMI,DM)
C ---------------------------------------------------------------------
C THIS SUBPROGRAM CALCULATES THE INVERSE OF DAMAGE EFFECT TENSOR M    I
C ---------------------------------------------------------------------
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION DMI(6,6),DM(6,6)
C
      DO 10 I=1,6
      DO 10 J=1,6
 10   DMI(I,J)=DM(I,J)
C
      CALL AIINV(DMI)
C
      RETURN
      END
C
C =====================================================================
C ====================== M D P H 2 =====================================
C =====================================================================
      SUBROUTINE MDPH2(XMPH2,PHI)
C ---------------------------------------------------------------------
C I THIS SUBPROGRAM CALCULATES THE DERIVATIVE OF DAMAGE EFFECT TENSOR I
C ---------------------------------------------------------------------
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION XMPH2(6,6,6,6),PHI(6),G1(6),F(6,6),F1(6,6,6)
      DIMENSION F2(6,6,6,6),G2(6,6)
C
      P1=1.D0-PHI(1)
      P2=1.D0-PHI(2)
      P3=1.D0-PHI(3)
      P4=PHI(5)*PHI(5)*P1
      P5=PHI(6)*PHI(6)*P2
      P6=PHI(4)*PHI(4)*P3
      P7=2.D0*PHI(4)*PHI(5)*PHI(6)
      G=2.D0*(P1*P2*P3-P4-P5-P6-P7)
      GG=G*G
      G4=GG*GG
C
      G1(1)=-2.D0*(P2*P3-PHI(5)**2)
      G1(2)=-2.D0*(P1*P3-PHI(6)**2)
      G1(3)=-2.D0*(P1*P2-PHI(4)**2)
      G1(4)=-4.D0*(PHI(4)*P3+PHI(5)*PHI(6))
      G1(5)=-4.D0*(PHI(5)*P1+PHI(4)*PHI(6))
      G1(6)=-4.D0*(PHI(6)*P2+PHI(4)*PHI(5))
C
      G2(1,2)=2.D0*P3
      G2(1,3)=2.D0*P2
      G2(1,5)=4.D0*PHI(5)
C
      G2(2,1)=2.D0*P3
      G2(2,3)=2.D0*P1
      G2(2,6)=4.D0*PHI(6)
C
      G2(3,1)=2.D0*P2
      G2(3,2)=2.D0*P1
      G2(3,4)=4.D0*PHI(4)
C
      G2(4,3)=4.D0*PHI(4)
      G2(4,4)=-4.D0*P3
      G2(4,5)=-4.D0*PHI(6)
      G2(4,6)=-4.D0*PHI(5)
C
      G2(5,1)=4.D0*PHI(5)
      G2(5,4)=-4.D0*PHI(6)
      G2(5,5)=-4.D0*P1
      G2(5,6)=-4.D0*PHI(4)
C
      G2(6,2)=4.D0*PHI(6)
      G2(6,4)=-4.D0*PHI(5)
      G2(6,5)=-4.D0*PHI(4)
      G2(6,6)=-4.D0*P2
C
      F(1,1)=2.D0*(P2*P3-PHI(5)*PHI(5))
      F(1,4)=2.D0*(PHI(6)*PHI(5)+PHI(4)*P3)
      F(1,6)=2.D0*(PHI(4)*PHI(5)+PHI(6)*P2)
C
```

138

```
      F(2,2)=2.D0*(P1*P3-PHI(6)*PHI(6))
      F(2,4)=F(1,4)
      F(2,5)=2.D0*(PHI(4)*PHI(6)+PHI(5)*P1)
C
      F(3,3)=2.D0*(P1*P2-PHI(4)*PHI(4))
      F(3,5)=F(2,5)
      F(3,6)=F(1,6)
C
      F(4,1)=0.5D0*F(1,4)
      F(4,2)=F(4,1)
      F(4,4)=P2*P3+P1*P3-PHI(5)*PHI(5)-PHI(6)*PHI(6)
      F(4,5)=0.5D0*F(1,6)
      F(4,6)=0.5D0*F(2,5)
C
      F(5,2)=0.5D0*F(2,5)
      F(5,3)=F(5,2)
      F(5,4)=0.5D0*F(1,6)
      F(5,5)=P1*P3+P1*P2-PHI(6)*PHI(6)-PHI(4)*PHI(4)
      F(5,6)=0.5D0*F(1,4)
C
      F(6,1)=F(5,4)
      F(6,3)=F(6,1)
      F(6,4)=F(5,2)
      F(6,5)=F(5,6)
      F(6,6)=P2*P3+P1*P2-PHI(5)*PHI(5)-PHI(4)*PHI(4)
C
C
      F1(1,1,2)=2.D0*(-P3)
      F1(1,1,3)=2.D0*(-P2)
      F1(1,1,5)=-4.D0*PHI(5)
C
      F1(1,4,3)=2.D0*(-PHI(4))
      F1(1,4,4)=2.D0*P3
      F1(1,4,5)=2.D0*PHI(6)
      F1(1,4,6)=2.D0*PHI(5)
C
      F1(1,6,2)=2.D0*(-PHI(6))
      F1(1,6,4)=2.D0*PHI(5)
      F1(1,6,5)=2.D0*PHI(4)
      F1(1,6,6)=2.D0*P2
C
C
      F1(2,2,1)=2.D0*(-P3)
      F1(2,2,3)=2.D0*(-P1)
      F1(2,2,6)=-4.D0*PHI(6)
C
      F1(2,4,3)=F1(1,4,3)
      F1(2,4,4)=F1(1,4,4)
      F1(2,4,5)=F1(1,4,5)
      F1(2,4,6)=F1(1,4,6)
C
      F1(2,5,1)=2.D0*(-PHI(5))
      F1(2,5,4)=2.D0*PHI(6)
      F1(2,5,5)=2.D0*P1
      F1(2,5,6)=2.D0*PHI(4)
C
C
      F1(3,3,1)=2.D0*(-P2)
      F1(3,3,2)=2.D0*(-P1)
      F1(3,3,4)=-4.D0*PHI(4)
C
      F1(3,5,1)=F1(2,5,1)
      F1(3,5,4)=F1(2,5,4)
      F1(3,5,5)=F1(2,5,5)
      F1(3,5,6)=F1(2,5,6)
C
      F1(3,6,2)=F1(1,6,2)
      F1(3,6,4)=F1(1,6,4)
      F1(3,6,5)=F1(1,6,5)
      F1(3,6,6)=F1(1,6,6)
C
C
      F1(4,1,3)=0.5D0*F1(1,4,3)
      F1(4,1,4)=0.5D0*F1(1,4,4)
      F1(4,1,5)=0.5D0*F1(1,4,5)
      F1(4,1,6)=0.5D0*F1(1,4,6)
C
      F1(4,2,3)=F1(4,1,3)
      F1(4,2,4)=F1(4,1,4)
      F1(4,2,5)=F1(4,1,5)
```

```
              F1(4,2,6)=F1(4,1,6)
      C

              F1(4,4,1)=-P3
              F1(4,4,2)=-P3
              F1(4,4,3)=-P2-P1
              F1(4,4,5)=-2.D0*PHI(5)
              F1(4,4,6)=-2.D0*PHI(6)
      C

              F1(4,5,2)=0.5D0*F1(1,6,2)
              F1(4,5,4)=0.5D0*F1(1,6,4)
              F1(4,5,5)=0.5D0*F1(1,6,5)
              F1(4,5,6)=0.5D0*F1(1,6,6)
      C

              F1(4,6,1)=0.5D0*F1(2,5,1)
              F1(4,6,4)=0.5D0*F1(2,5,4)
              F1(4,6,5)=0.5D0*F1(2,5,5)
              F1(4,6,6)=0.5D0*F1(2,5,6)
      C
      C

              F1(5,2,1)=0.5D0*F1(2,5,1)
              F1(5,2,4)=0.5D0*F1(2,5,4)
              F1(5,2,5)=0.5D0*F1(2,5,5)
              F1(5,2,6)=0.5D0*F1(2,5,6)
      C

              F1(5,3,1)=F1(5,2,1)
              F1(5,3,4)=F1(5,2,4)
              F1(5,3,5)=F1(5,2,5)
              F1(5,3,6)=F1(5,2,6)
      C

              F1(5,4,2)=0.5D0*F1(1,6,2)
              F1(5,4,4)=0.5D0*F1(1,6,4)
              F1(5,4,5)=0.5D0*F1(1,6,5)
              F1(5,4,6)=0.5D0*F1(1,6,6)
      C

              F1(5,5,1)=-P3-P2
              F1(5,5,2)=-P1
              F1(5,5,3)=-P1
              F1(5,5,4)=-2.D0*PHI(4)
              F1(5,5,6)=-2.D0*PHI(6)
      C

              F1(5,6,3)=0.5D0*F1(1,4,3)
              F1(5,6,4)=0.5D0*F1(1,4,4)
              F1(5,6,5)=0.5D0*F1(1,4,5)
              F1(5,6,6)=0.5D0*F1(1,4,6)
      C
      C

              F1(6,1,2)=F1(5,4,2)
              F1(6,1,4)=F1(5,4,4)
              F1(6,1,5)=F1(5,4,5)
              F1(6,1,6)=F1(5,4,6)
      C

              F1(6,3,2)=F1(6,1,2)
              F1(6,3,4)=F1(6,1,4)
              F1(6,3,5)=F1(6,1,5)
              F1(6,3,6)=F1(6,1,6)
      C

              F1(6,4,1)=F1(5,2,1)
              F1(6,4,4)=F1(5,2,4)
              F1(6,4,5)=F1(5,2,5)
              F1(6,4,6)=F1(5,2,6)
      C

              F1(6,5,3)=F1(5,6,3)
              F1(6,5,4)=F1(5,6,4)
              F1(6,5,5)=F1(5,6,5)
              F1(6,5,6)=F1(5,6,6)
      C
              F1(6,6,1)=-P2
              F1(6,6,2)=-P3-P1
              F1(6,6,3)=-P2
              F1(6,6,4)=-2.D0*PHI(4)
              F1(6,6,5)=-2.D0*PHI(5)
      C
      C
      C
              F2(1,1,2,3)=2.D0
              F2(1,1,3,2)=2.D0
              F2(1,1,5,5)=-4.D0
      C

              F2(1,4,3,4)=-2.D0
              F2(1,4,4,3)=-2.D0
```

140

```
      F2(1,4,5,6)=2.D0
      F2(1,4,6,5)=2.D0
C
      F2(1,6,2,6)=-2.D0
      F2(1,6,4,5)=2.D0
      F2(1,6,5,4)=2.D0
      F2(1,6,6,2)=-2.D0
C
C
      F2(2,2,1,3)=2.D0
      F2(2,2,3,1)=2.D0
      F2(2,2,6,6)=-4.D0
C
      F2(2,4,3,4)=F2(1,4,3,4)
      F2(2,4,4,3)=F2(1,4,4,3)
      F2(2,4,5,6)=F2(1,4,5,6)
      F2(2,4,6,5)=F2(1,4,6,5)
C
      F2(2,5,1,5)=-2.D0
      F2(2,5,4,6)=2.D0
      F2(2,5,5,1)=-2.D0
      F2(2,5,6,4)=2.D0
C
C
      F2(3,3,1,2)=2.D0
      F2(3,3,2,1)=2.D0
      F2(3,3,4,4)=-4.D0
C
      F2(3,5,1,5)=F2(2,5,1,5)
      F2(3,5,4,6)=F2(2,5,4,6)
      F2(3,5,5,1)=F2(2,5,5,1)
      F2(3,5,6,4)=F2(2,5,6,4)
C
      F2(3,6,2,6)=F2(1,6,2,6)
      F2(3,6,4,5)=F2(1,6,4,5)
      F2(3,6,5,4)=F2(1,6,5,4)
      F2(3,6,6,2)=F2(1,6,6,2)
C
C
      F2(4,1,3,4)=0.5D0*F2(1,4,3,4)
      F2(4,1,4,3)=0.5D0*F2(1,4,4,3)
      F2(4,1,5,6)=0.5D0*F2(1,4,5,6)
      F2(4,1,6,5)=0.5D0*F2(1,4,6,5)
C
      F2(4,2,3,4)=F2(4,1,3,4)
      F2(4,2,4,3)=F2(4,1,4,3)
      F2(4,2,5,6)=F2(4,1,5,6)
      F2(4,2,6,5)=F2(4,1,6,5)
C
      F2(4,4,1,3)=1.D0
      F2(4,4,2,3)=1.D0
      F2(4,4,3,1)=1.D0
      F2(4,4,3,2)=1.D0
      F2(4,4,5,5)=-2.D0
      F2(4,4,6,6)=-2.D0
C
      F2(4,5,2,6)=0.5D0*F2(1,6,2,6)
      F2(4,5,4,5)=0.5D0*F2(1,6,4,5)
      F2(4,5,5,4)=0.5D0*F2(1,6,5,4)
      F2(4,5,6,2)=0.5D0*F2(1,6,6,2)
C
      F2(4,6,1,5)=0.5D0*F2(2,5,1,5)
      F2(4,6,4,6)=0.5D0*F2(2,5,4,6)
      F2(4,6,5,1)=0.5D0*F2(2,5,5,1)
      F2(4,6,6,4)=0.5D0*F2(2,5,6,4)
C
C
      F2(5,2,1,5)=0.5D0*F2(2,5,1,5)
      F2(5,2,4,6)=0.5D0*F2(2,5,4,6)
      F2(5,2,5,1)=0.5D0*F2(2,5,5,1)
      F2(5,2,6,4)=0.5D0*F2(2,5,6,4)
C
      F2(5,3,1,5)=F2(5,2,1,5)
      F2(5,3,4,6)=F2(5,2,4,6)
      F2(5,3,5,1)=F2(5,2,5,1)
      F2(5,3,6,4)=F2(5,2,6,4)
C
      F2(5,4,2,6)=0.5D0*F2(1,6,2,6)
      F2(5,4,4,5)=0.5D0*F2(1,6,4,5)
      F2(5,4,5,4)=0.5D0*F2(1,6,5,4)
```

```fortran
      F2(5,4,6,2)=0.5D0*F2(1,6,6,2)
C
      F2(5,5,1,2)=1.D0
      F2(5,5,1,3)=1.D0
      F2(5,5,2,1)=1.D0
      F2(5,5,3,1)=1.D0
      F2(5,5,4,4)=-2.D0
      F2(5,5,6,6)=-2.D0
C
      F2(5,6,3,4)=0.5D0*F2(1,4,3,4)
      F2(5,6,4,3)=0.5D0*F2(1,4,4,3)
      F2(5,6,5,6)=0.5D0*F2(1,4,5,6)
      F2(5,6,6,5)=0.5D0*F2(1,4,6,5)
C
C
      F2(6,1,2,6)=F2(5,4,2,6)
      F2(6,1,4,5)=F2(5,4,4,5)
      F2(6,1,5,4)=F2(5,4,5,4)
      F2(6,1,6,2)=F2(5,4,6,2)
C
      F2(6,3,2,6)=F2(6,1,2,6)
      F2(6,3,4,5)=F2(6,1,4,5)
      F2(6,3,5,4)=F2(6,1,5,4)
      F2(6,3,6,2)=F2(6,1,6,2)
C
      F2(6,4,1,5)=F2(5,2,1,5)
      F2(6,4,4,6)=F2(5,2,4,6)
      F2(6,4,5,1)=F2(5,2,5,1)
      F2(6,4,6,4)=F2(5,2,6,4)
C
      F2(6,5,3,4)=F2(5,6,3,4)
      F2(6,5,4,3)=F2(5,6,4,3)
      F2(6,5,5,6)=F2(5,6,5,6)
      F2(6,5,6,5)=F2(5,6,6,5)
C
      F2(6,6,1,2)=1.D0
      F2(6,6,2,1)=1.D0
      F2(6,6,2,3)=1.D0
      F2(6,6,3,2)=1.D0
      F2(6,6,4,4)=-2.D0
      F2(6,6,5,5)=-2.D0
C
      DO 10 I=1,6
      DO 10 J=1,6
      DO 10 K=1,6
      DO 10 L=1,6
   10 XMPH2(I,J,K,L)=((F2(I,J,K,L)*G+F1(I,J,K)*G1(L)-F1(I,J,L)*G1(K)
     $-F(I,J)*G2(K,L))*GG-(F1(I,J,K)*G-F(I,J)*G1(K))*2.D0*G*G1(L))/G4
C
      RETURN
      END
```